|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ *Робототехника и комплексная автоматизация*

КАФЕДРА *Системы автоматизированного проектирования (РК-6)*

**ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

по курсу: «Базы данных»

|  |  |  |
| --- | --- | --- |
| Студент |  | Жарова Татьяна Александровна |
| Группа |  | РК6-45Б |
| Тип задания |  | Лабораторная работа |

Студент **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Жарова Т.А.**

*подпись, дата                   фио.*

Преподаватель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_ Пивоварова Н.В.**

*подпись, дата                   фио.*

Оценка  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Москва, 2022 г.*

**Оглавление**

[Техническое задание: 3](#_Toc103177508)

[Инфологическая модель: 4](#_Toc103177509)

[Логическая модель: 4](#_Toc103177510)

[Простые запросы: 5](#_Toc103177511)

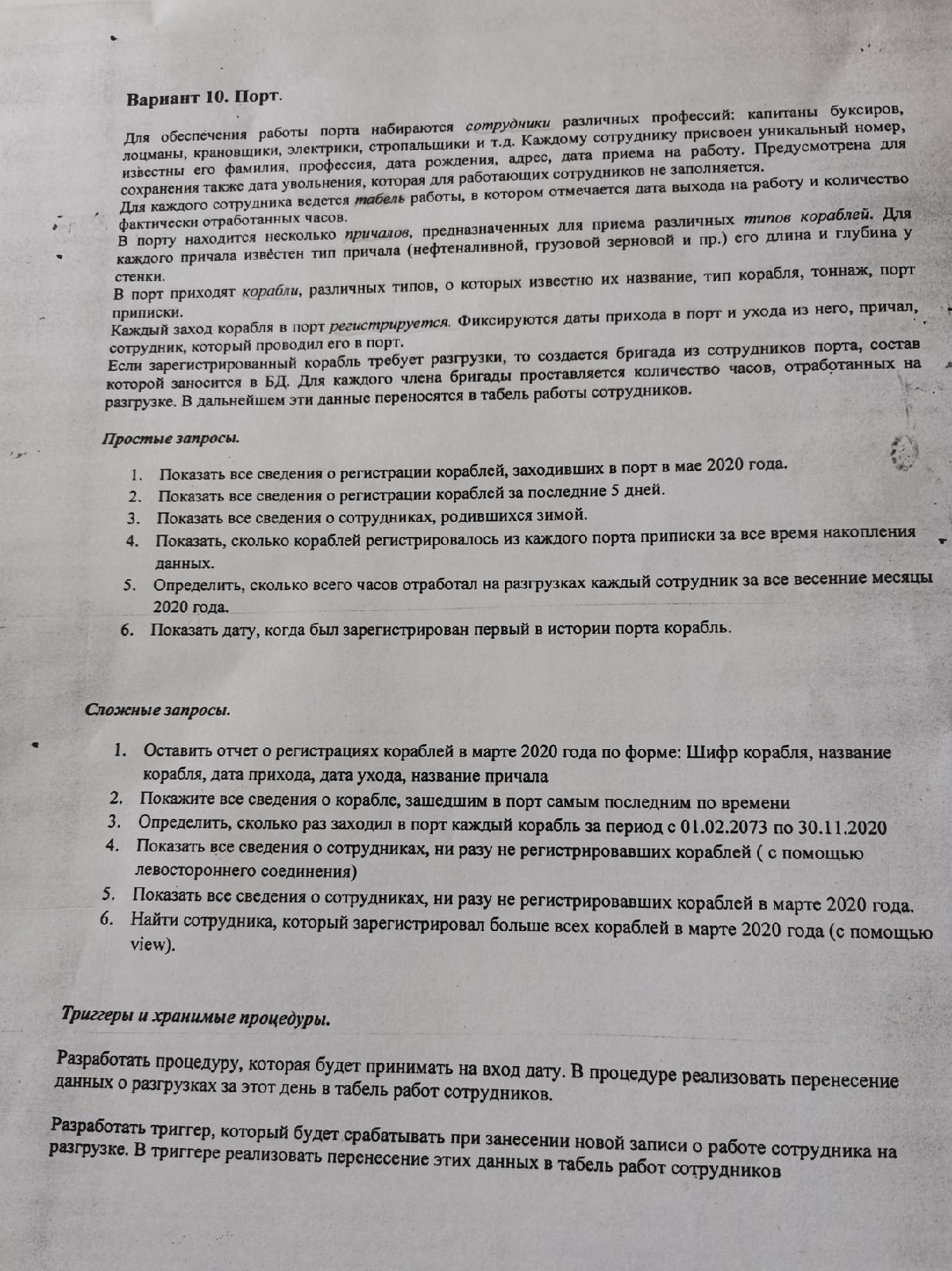
[Сложные запросы: 8](#_Toc103177512)

[План процедуры: 11](#_Toc103177513)

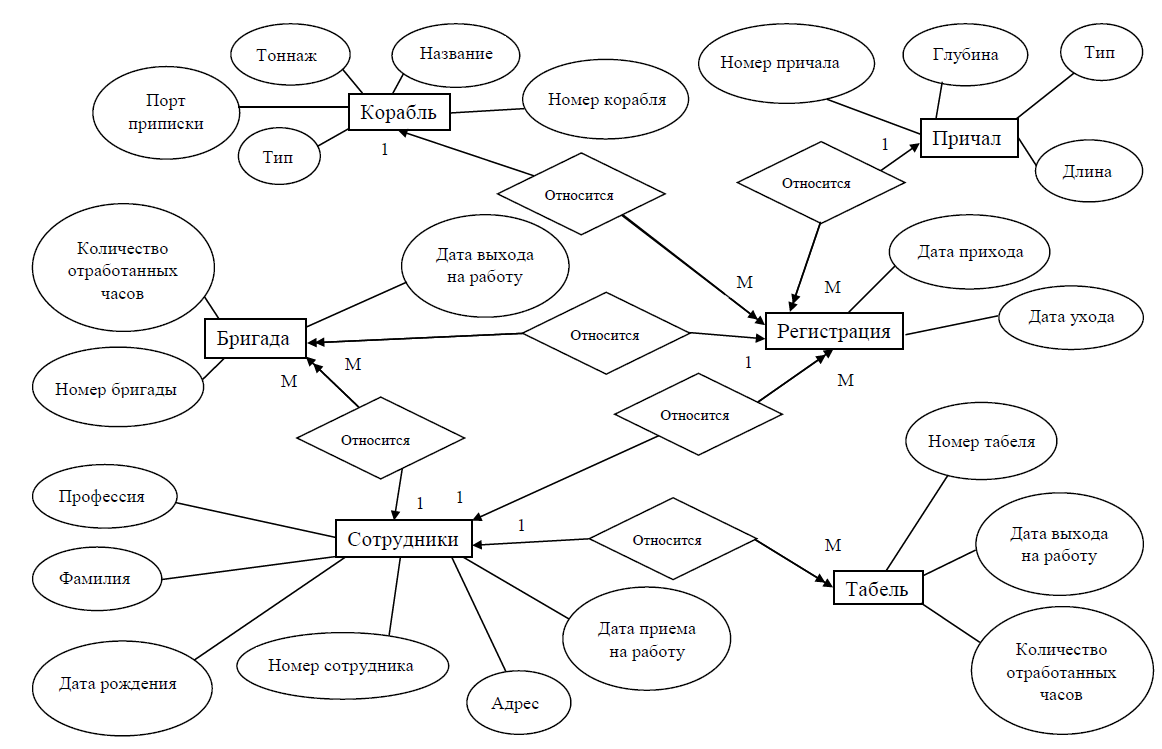
[Процедура 11](#_Toc103177514)

[Триггер 13](#_Toc103177515)

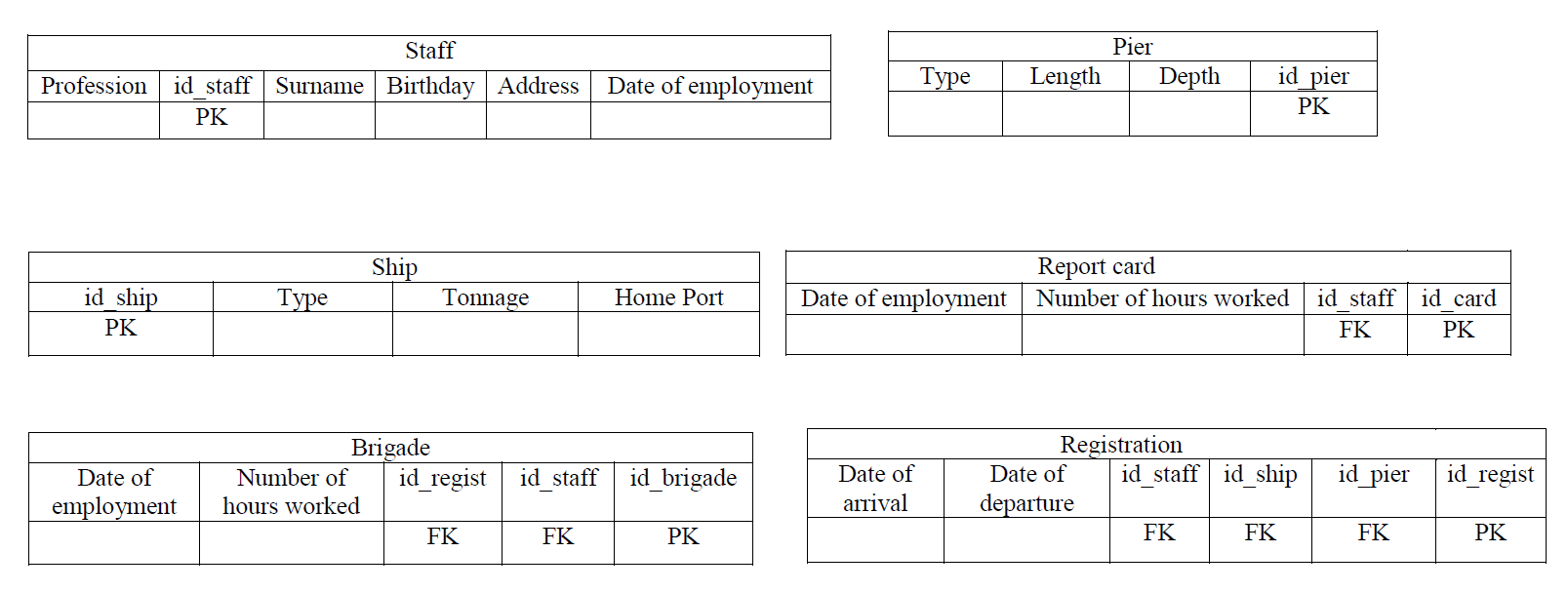
# **Техническое задание:**



# **Инфологическая модель:**



# **Логическая модель:**

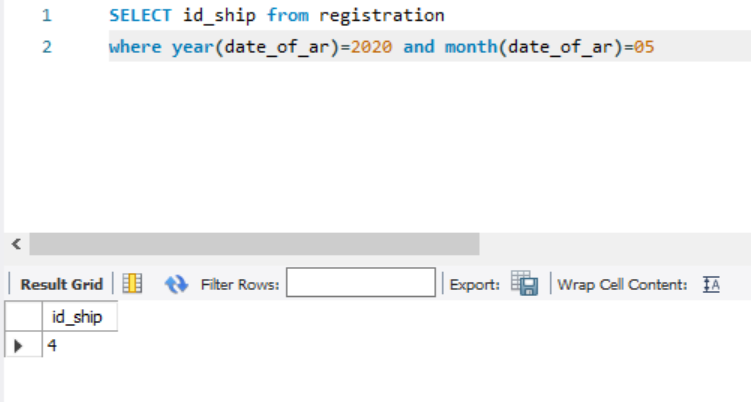


# **Простые запросы:**

1) Показать все сведения о регистрации кораблей, заходивших в мае 2020 года:

SELECT id\_ship from registration

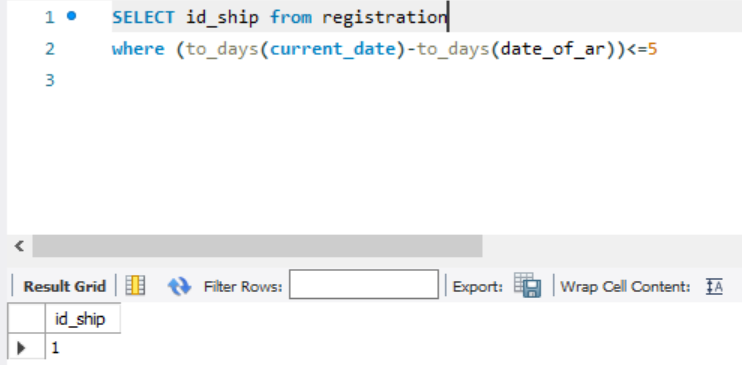
where year(date\_of\_ar)=2020 and month(date\_of\_ar)=05



2) Показать все сведения о регистрации кораблей за последние 5 дней:

SELECT id\_ship from registration

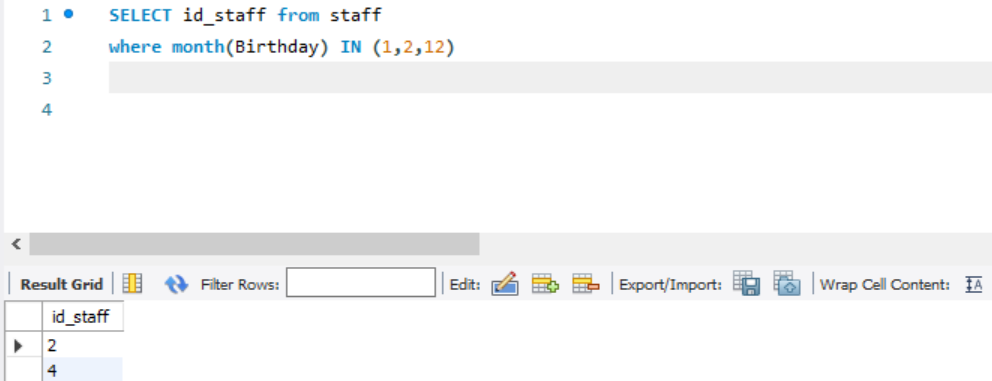
where (to\_days(current\_date)-to\_days(date\_of\_ar))<=5



3) Показать все сведения о сотрудниках, родившихся зимой:

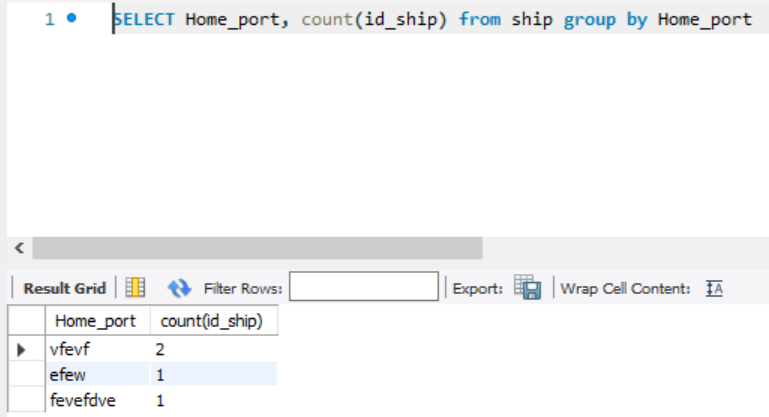
SELECT id\_staff from staff

where month(Birthday) IN (1,2,12)



4) Показать, сколько кораблей регистрировалось из каждого порта приписки за все время накопления данных:

SELECT Home\_port, count(id\_ship) from ship group by Home\_port

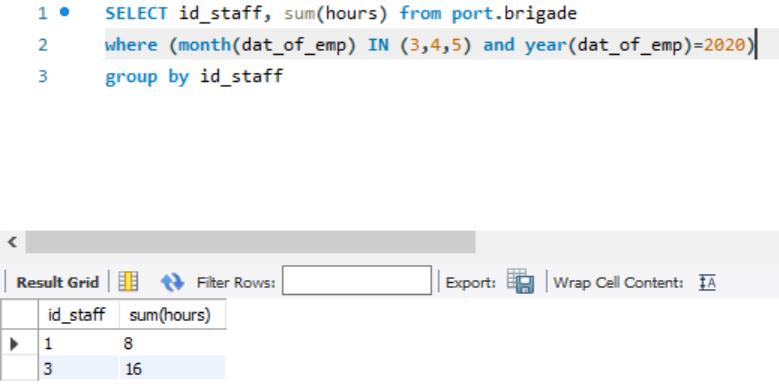


5) Определить, сколько всего часов отработал на разгрузках каждый сотрудник за все весенние месяцы 2020 года:

SELECT id\_staff, sum(hours) from port.brigade

where (month(dat\_of\_emp) IN (3,4,5) and year(dat\_of\_emp)=2020)

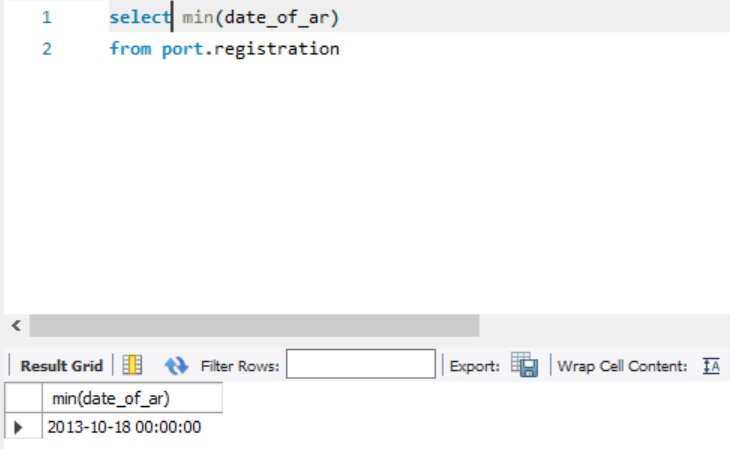
group by id\_staff



6) Показать дату, когда был зарегистрирован первый в истории порта корабль:

select min(date\_of\_ar)

from port.registration



# **Сложные запросы:**

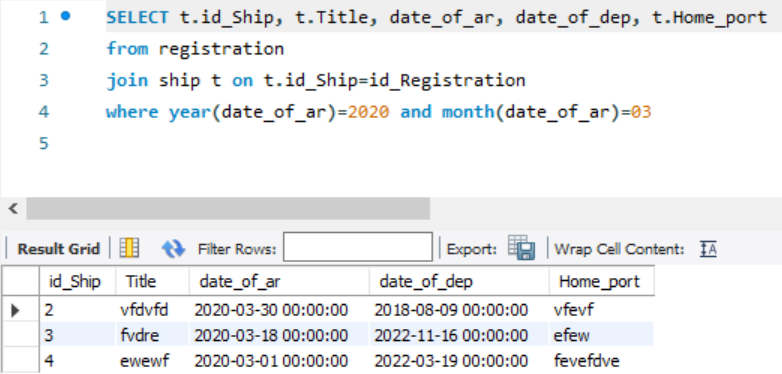
1) Оставить отчет о регистрации кораблей в марте 2020 года по форме: Шифр корабля, название корабля, дата прихода и ухода, название причала:

SELECT t.id\_Ship, t.Title, date\_of\_ar, date\_of\_dep, t.Home\_port

from registration

join ship t on t.id\_Ship=id\_Registration

where year(date\_of\_ar)=2020 and month(date\_of\_ar)=03

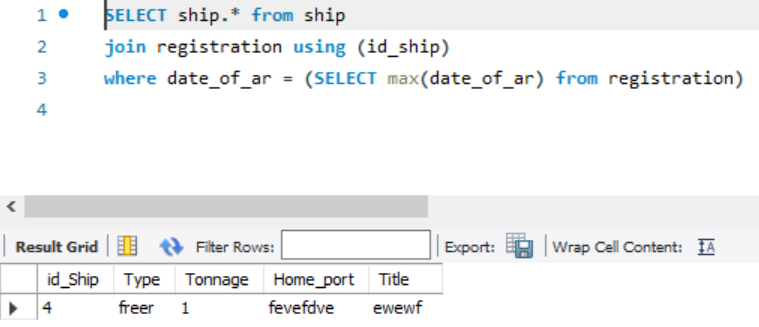


2) Покажите все сведения о корабле, зашедшим в порт самым последним по времени:

SELECT ship.\* from ship

join registration using (id\_ship)

where date\_of\_ar = (SELECT max(date\_of\_ar) from registration)



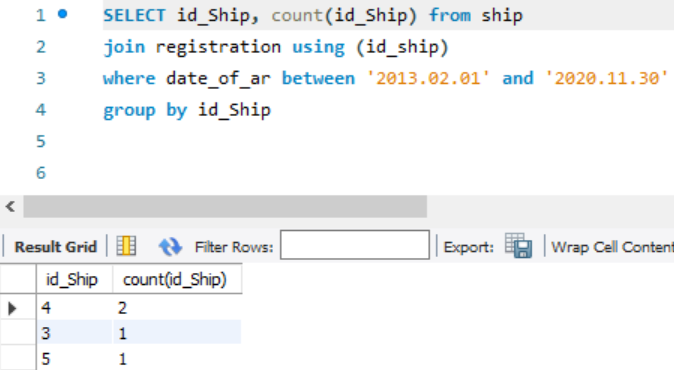
3) Определить, сколько раз заходил в порт каждый корабль за период с 01.02.2013 по 30.11.2020:

SELECT id\_Ship, count(id\_Ship) from ship

join registration using (id\_ship)

where date\_of\_ar between '2013.02.01' and '2020.11.30'

group by id\_Ship

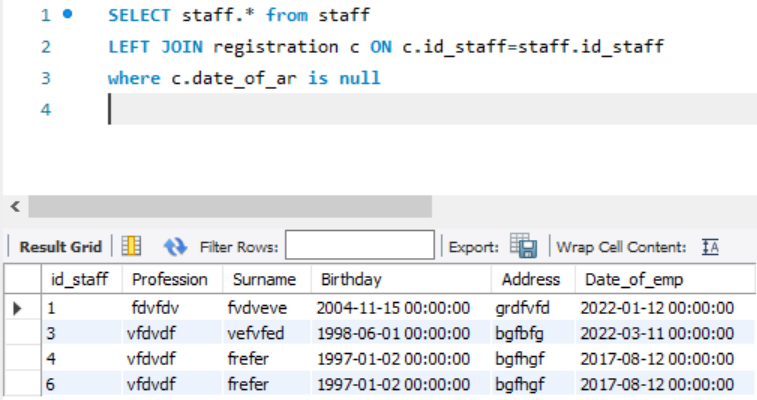


4) Показать все сведения о сотрудниках, ни разу не регистрировавших кораблей

SELECT staff.\* from staff

LEFT JOIN registration c ON c.id\_staff=staff.id\_staff

where c.date\_of\_ar is null

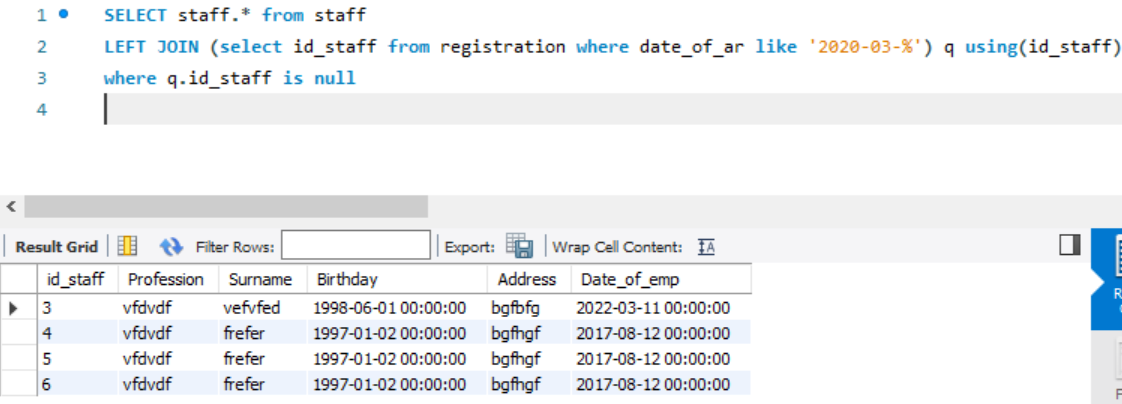


5) Показать все сведения о сотрудниках, ни разу не регистрировавших кораблей в марте 2020 года

SELECT staff.\* from staff

LEFT JOIN (select id\_staff from registration where date\_of\_ar like '2020-03-%') q using(id\_staff)

where q.id\_staff is null



6) Найти сотрудника, который зарегистрировал больше всех кораблей в марте 2020 года

CREATE VIEW v(id\_staff,count) AS select id\_staff, count(id\_ship)

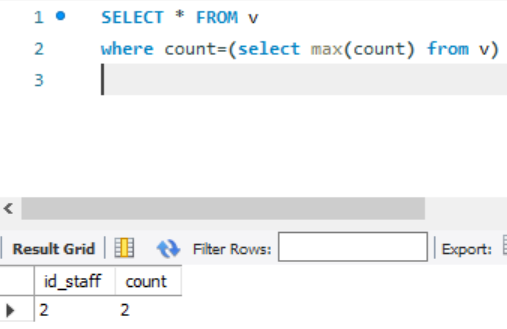
from registration

where date\_of\_ar like '2020-03-%'

group by id\_staff

SELECT \* FROM v

where count=(select max(count) from v)



# **План процедуры:**

Разработать процедуру, которая будет принимать на вход дату. В процедуре реализовать перенесение данных о разгрузках за этот день в табель работ сотрудников

1. Декларируем вспомогательные переменные e\_num и hours\_num, означающие соответственно id сотрудника и количество отработанных им часов на разгрузке, а также done типа integer, характеризующая состояние цикла;
2. Декларируем курсор, в который заносятся id сотрудника и количество отработанных им часов на разгрузке в веденную пользователем дату;

DECLARE C1 CURSOR FOR

SELECT id\_staff, hours FROM brigade

WHERE dat\_of\_emp = usersdate;

1. Декларируем обработчик для отслеживания конца курсора;

DECLARE exit handler for SQLSTATE '02000' set done = 1;

1. Открываем курсор;
2. Открываем цикл чтения записей из курсора;

fetch C1 into e\_num, hours\_num;

1. Каждая запись о работе сотрудника на разгрузке, подходящая по дате, переносится в табель работы сотрудников;

INSERT report\_card VALUES(NULL, e\_num, usersdate, hours\_num);

1. По достижению конца курсора производится выход из цикла и последующее закрытие курсора.

# **Процедура**

Разработать процедуру, которая будет принимать на вход дату. В процедуре реализовать перенесение данных о разгрузках за этот день в табель работ сотрудников

CREATE DEFINER=`root`@`localhost` PROCEDURE `user\_unload`(usersdate DATE)

BEGIN

DECLARE e\_num, hours\_num integer;

DECLARE done integer DEFAULT 0;

DECLARE C1 CURSOR FOR

SELECT id\_staff, hours FROM brigade

WHERE dat\_of\_emp = usersdate;

DECLARE exit handler for SQLSTATE '02000' set done = 1;

OPEN C1;

WHILE done = 0 DO

fetch C1 into e\_num, hours\_num;

INSERT report\_card VALUES(NULL, e\_num, usersdate, hours\_num);

END WHILE;

CLOSE C1;

END

Таблица бригады:

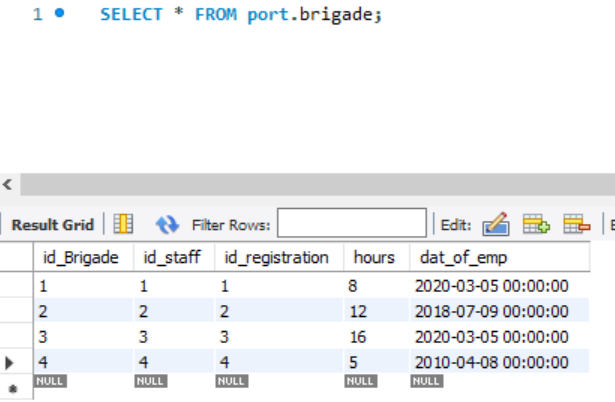


Таблица табеля работ до выполнения процедуры:

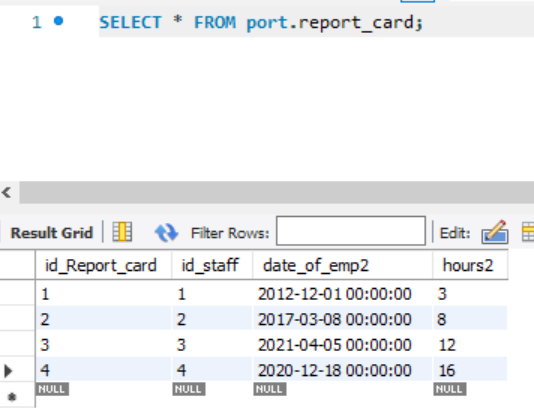
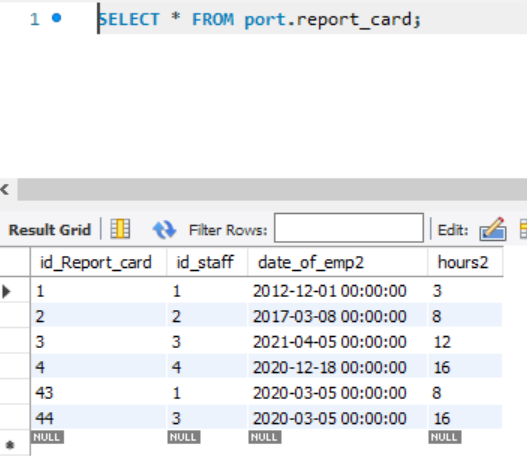


Таблица табеля работ после выполнения процедуры:

CALL user\_unload('2020-03-05');



# **Триггер**

Разработать триггер, который будет срабатывать при занесении новой записи о работе сотрудника на разгрузке. В триггере реализовать перенесение этих данных в табель работ сотрудников

CREATE DEFINER=`root`@`localhost` TRIGGER `brigade\_AFTER\_INSERT` AFTER INSERT ON `brigade` FOR EACH ROW BEGIN

DECLARE e\_num, hours\_num integer;

DECLARE dateu DATE;

DECLARE done integer DEFAULT 0;

DECLARE C1 CURSOR FOR

SELECT hours, id\_staff, dat\_of\_emp FROM brigade

WHERE id\_staff = new.id\_staff;

DECLARE exit HANDLER FOR SQLSTATE '02000' SET done = 1;

OPEN C1;

WHILE done = 0 DO

fetch C1 into hours\_num, e\_num, dateu;

INSERT report\_card VALUES(NULL, e\_num, dateu, hours\_num);

END WHILE;

CLOSE C1;

END

Таблица бригады до добавления новой записи:

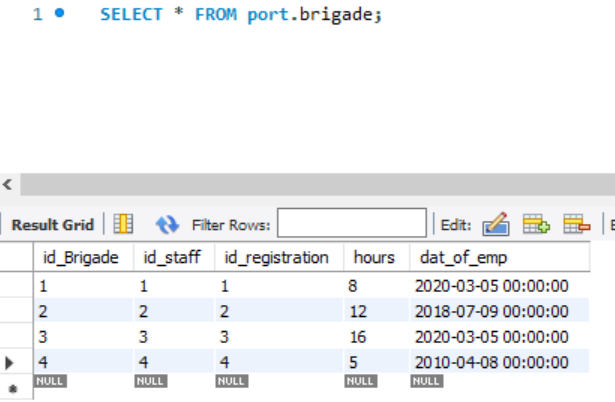


Таблица табеля работ до выполнения триггера:

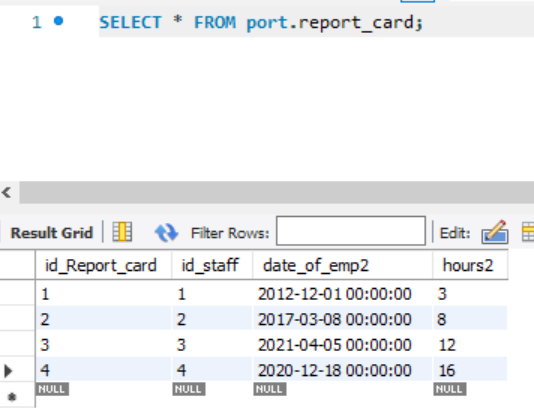


Таблица бригады после добавления новой записи:

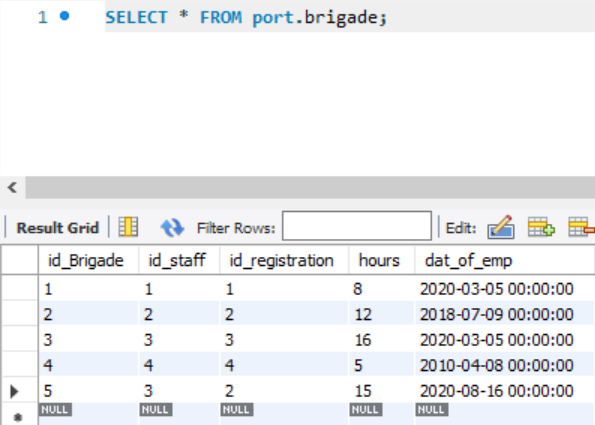


Таблица табеля работ после выполнения триггера:

