

# ГУАП

## ФАКУЛЬТЕТ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

ОТЧЁТ  
ЗАЩИЩЁН С ОЦЕНКОЙ  
ПРЕПОДАВАТЕЛЬ

<u>преподаватель</u>	<u></u>	<u>И. А. Юрьева</u>
должность, уч. степень, звание	подпись, дата	инициалы, фамилия

## ОТЧЁТЫ О ЛАБОРАТОРНЫХ РАБОТАХ

по дисциплине: МДК 04.01

РАБОТУ ВЫПОЛНИЛ  
СТУДЕНТ ГР. №

<u>С021</u>	<u></u>	<u>С. С. Гамуйло</u>
	подпись, дата	инициалы, фамилия

## СОДЕРЖАНИЕ

Лабораторная работа №3 .....	3
------------------------------	---

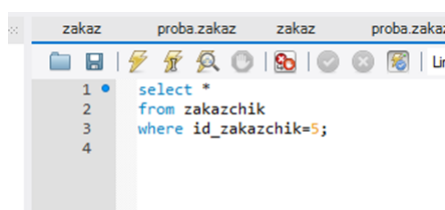
## Лабораторная работа №3

**Тема:** Создание хранимых процедур и функций в MySQL Workbench.  
Методы защиты баз данных.

**Цель работы:** получение практических навыков при создании и использовании хранимых процедур и функций.

### Самостоятельная работа

**Задание 1.** Создайте хранимую процедуру с параметром для выполнения следующего запроса.



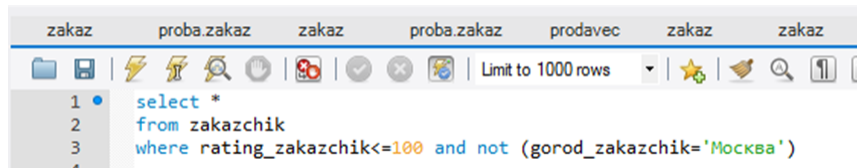
### Код процедуры

```
create procedure `request_id`(customer int)
begin
    select *
    from customer
    where customer.customer_id = customer;
end;
```

	customer_id	customer_name	city	rating	credit
1	3	Берегов Владимир Евгеньевич	Москва	96	500000

Рисунок 1 - Результат работы первой процедуры (id=3)

**Задание 2.** Создайте хранимую процедуру с параметрами для выполнения следующего запроса.



Код процедуры

```

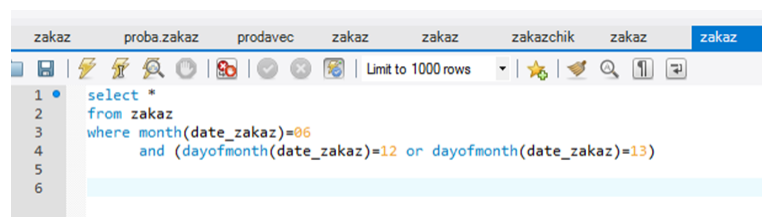
create procedure `request_city_rating`(rate int, city_not varchar(50))
begin
    select *
    from customer
    where rating <= rate and not(city = city_not);
end;

```

	customer_id	customer_name	city	rating	credit
1	2	Антипов Валерий Петрович	Екатеринбург	45	75000

Рисунок 2 - Результат работы процедуры (Не Москва и рейтинг не выше 50)

**Задание 3.** Создайте хранимую процедуру с параметрами для выполнения следующего запроса.



Код процедуры

```

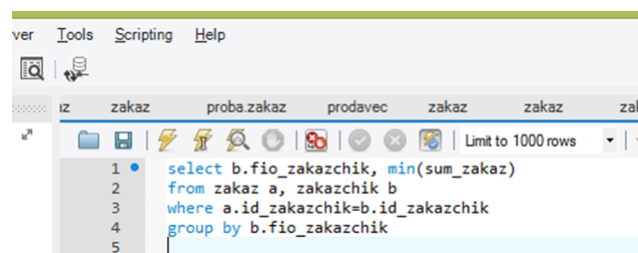
create procedure `request_date`(date date)
begin
    select *
    from `order`
    where month(order_date) = month(date)
    and (dayofmonth(order_date) = dayofmonth(date) or dayofmonth(order_date) =
dayofmonth(date) + 1);
end;

```

	order_id	order_total	order_date	seller_id	customer_id
1	4	60000	2010-06-13 00:00:00	5	1
2	5	35000	2010-06-14 00:00:00	4	5
3	6	70000	2010-06-14 00:00:00	4	3
4	7	65000	2010-06-14 00:00:00	2	1
5	8	50000	2010-06-14 00:00:00	2	5

Рисунок 3 - Результат работы процедуры (дата 2010-06-13)

**Задание 4.** Создайте хранимую процедуру с параметром, которая в зависимости от параметра находит наименьшую или наибольшую сумму для каждого заказчика.



Код процедуры

```

create procedure `min_or_max_request`(minmax bool)
begin
    if(minmax) then select customer.customer_name, min(order_total)
        from customer, `order`
        where `order`.customer_id = customer.customer_id;
    else select customer.customer_name, max(order_total)
        from customer, `order`
        where `order`.customer_id = customer.customer_id;
    end if;
end;

```

	customer_name	min(order_total)
1	Антипов Валерий Петрович	30000
2	Петров Олег Дмитриевич	60000
3	Голубев Иосиф Николаевич	35000
4	Берегов Владимир Евгеньевич	70000

Рисунок 4 - Результат работы процедуры (параметр = 1) (минимально)

**Задание 5.** Создайте хранимую процедуру с параметром для выполнения следующего запроса.

```

update prodavec
set comis_prod=comis_prod*1.2
where plan_prod>300

```

Код процедуры

```

create
definer = root@localhost procedure update_seller(IN sales int, IN multiply
double)
begin
    update seller
    set commission = commission * multiply
    where sales_plan > sales;
end;

```

commission	sales_plan
18	150
172	300
18	100
119	200

Рисунок 5 - Результат работы процедуры (вторая и третья строки умножились на 3.5)

**Задание 6.** Создайте хранимую процедуру с параметром для выполнения следующего запроса.

Код процедуры

```
create
  definer = root@localhost procedure delete_order(IN name varchar(100))
begin
  delete from `order`
    where customer_id = (select customer_id
                        from customer
                        where customer_name = name);
end;
```

**Индивидуальное задание**

**Процедура 1.** Получить всех пассажиров всех рейсов определенной даты

Код процедуры

```

create procedure getFlightsByDate(date date)
begin
    select f.flightCode, f.date, f.from, f.to, a.brand, a.capacity, a.load,
    t.passFirstName, t.passSecondName, t.passMiddleName, t.price, t.place
    from flights f
    join aircrafts a on f.aircraftCode = a.aircraftCode
    join tickets t on f.flightCode = t.flightCode
    where YEAR(f.date) = YEAR(date) and MONTH(f.date) = MONTH(date) and
DAYOFMONTH(f.date);
end;

```

	load	passFirstName	passSecondName	passMiddleName	price	place
1	0	10290	Алексей	Шершавин	Саидович	920.01 C21
2	0	10290	Неформал	Арсеньев	Андреевич	900.01 A33

Рисунок 6 - Результат работы с датой 2020-12-03

## Процедура 2. Получить всех сотрудников на конкретной должности

Код процедуры

```

create procedure getEmployeesByJobTitle(jobTitle varchar(50))
begin
    select e.empCode, e.firstName, e.secondName, e.middleName, e.birthDate, e.sex,
    e.homeAddress, e.phoneNumber, e.passportDetails, j.title as jobTitle, c.crewCode
    from employee e
    join jobs j on e.jobCode = j.jobCode
    left join crewemployees c on e.empCode = c.crewMember
    where j.title = jobTitle;
end;

```

	sex	homeAddress	phoneNumber	passportDetails	jobTitle	crewCode
1		ул Пожарская, д. 120	28503099	8770153060	Второй пилот	1

Рисунок 7 - Результат с параметром Второй пилот

## Процедура 3. Обновить дату последнего обслуживания по коду самолета



### Код процедуры

```
create procedure updateAircraftReleaseDate(acCode int, lastService date)
begin
    update aircrafts
        set lastServiceDate = lastService
    where aircraftCode = acCode;
end;
```

lastServiceDate
2013-09-12
2022-11-18
2020-12-31
2021-10-29

Рисунок 8 - Результат работы - с 2019 на 2020 год

**Процедура 4.** Удалить всю информацию о самолете из всех таблиц с помощью транзакции

### Код процедуры

```
create
    definer = root@localhost procedure deleteAircraft(IN acCode int)
begin
    start transaction;

    delete from flights where aircraftCode = acCode;

    delete from tickets where flightCode in (select flightCode from flights where
        aircraftCode = acCode);

    delete from aircrafts where aircraftCode = acCode;

    if (select row_count()) = 0 then
        rollback;
    else
        commit;
    end if;
end;
```

## Процедура 5. Добавить нового сотрудника

### Код процедуры

```
create
definer = root@localhost procedure insertEmployee(IN empCode1 int, IN firstName1
varchar(50), IN secondName1 varchar(50), IN middleName1 varchar(50), IN birthDate1
date, IN sex1 char, IN homeAddress1 varchar(100), IN phoneNumber1 varchar(20), IN
passportDetails1 varchar(50), IN jobCode1 int)
begin
insert into employee values(empCode1, firstName1,
secondName1, middleName1,
birthDate1, sex1,
homeAddress1, phoneNumber1,
passportDetails1, jobCode1);
end;
```

**Представление 1.** Создать вид всей информации о самолете прикрепленном к конкретному рейсу

### Код представления

```
create view flightDetails as
SELECT flights.*, aircrafttypes.title AS aircraftType, aircrafttypes.restrictions
AS aircraftRestrictions
FROM flights
JOIN aircrafts ON flights.aircraftCode = aircrafts.aircraftCode
JOIN aircrafttypes ON aircrafts.typeCode = aircrafttypes.typeCode;
```

	flightCode	date	from	to
1	17	2023-02-13 00:00:00	New York	London
2	7	2022-02-11 08:09:31	Иваново	Пупинск
3	2	2020-12-03 11:46:27	Санкт-Петербург	Якутск

Рисунок 9 - Результат работы запроса

**Представление 2.** Отобразить информацию о том, сколько налетано часов у каждого типа самолетов

Код представления

```
CREATE VIEW aircraft_type_hours AS
SELECT aircrafttypes.title AS aircraftType, SUM(aircrafts.hoursFlown) AS
totalHoursFlown
FROM aircrafts
JOIN aircrafttypes ON aircrafts.typeCode = aircrafttypes.typeCode
GROUP BY aircrafttypes.title;
```



	 aircraftType	 totalHoursFlown
1	Грузовой	2660
2	Пассажирский	510

Рисунок 10 - Результат работы запроса