

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра МО ЭВМ**

**ОТЧЕТ**  
**по лабораторной работе №2**  
**по дисциплине «Базы данных»**  
**Тема: «Реализация базы данных в СУБД MySQL»**

Студент гр. 9383

\_\_\_\_\_

Гордон Д.А.

Преподаватель

\_\_\_\_\_

Заславский М.М.

Санкт-Петербург

2021

### **Цель работы.**

Реализовать БД в СУБД MySQL. Написать запросы, отвечающие на вопросы из ЛР1.

### **Задание.**

#### **Вариант 3.**

Написать запросы для создания таблиц из предыдущей лабораторной работы

Заполнить тестовыми данными: 5-10 строк на каждую таблицу, обязательно наличие связи между ними, данные приближены к реальности.

Написать запросы к БД, отвечающие на вопросы из предыдущей лабораторной работы:

- Какой предмет будет в заданном классе, в заданный день недели на заданном уроке?
- Кто из учителей преподает в заданном классе?
- В каком кабинете будет 5-й урок в среду у некоторого класса?
- В каких классах преподает заданный предмет заданный учитель?
- Расписание на заданный день недели для указанного класса?
- Сколько учеников в указанном классе?

## Выполнение работы.

Были написаны запросы по созданию таблиц в БД (см. приложение).

Таблицы БД были заполнены тестовыми данными (см. в приложении последние строки).

В итоге получилась такие таблицы и связи:

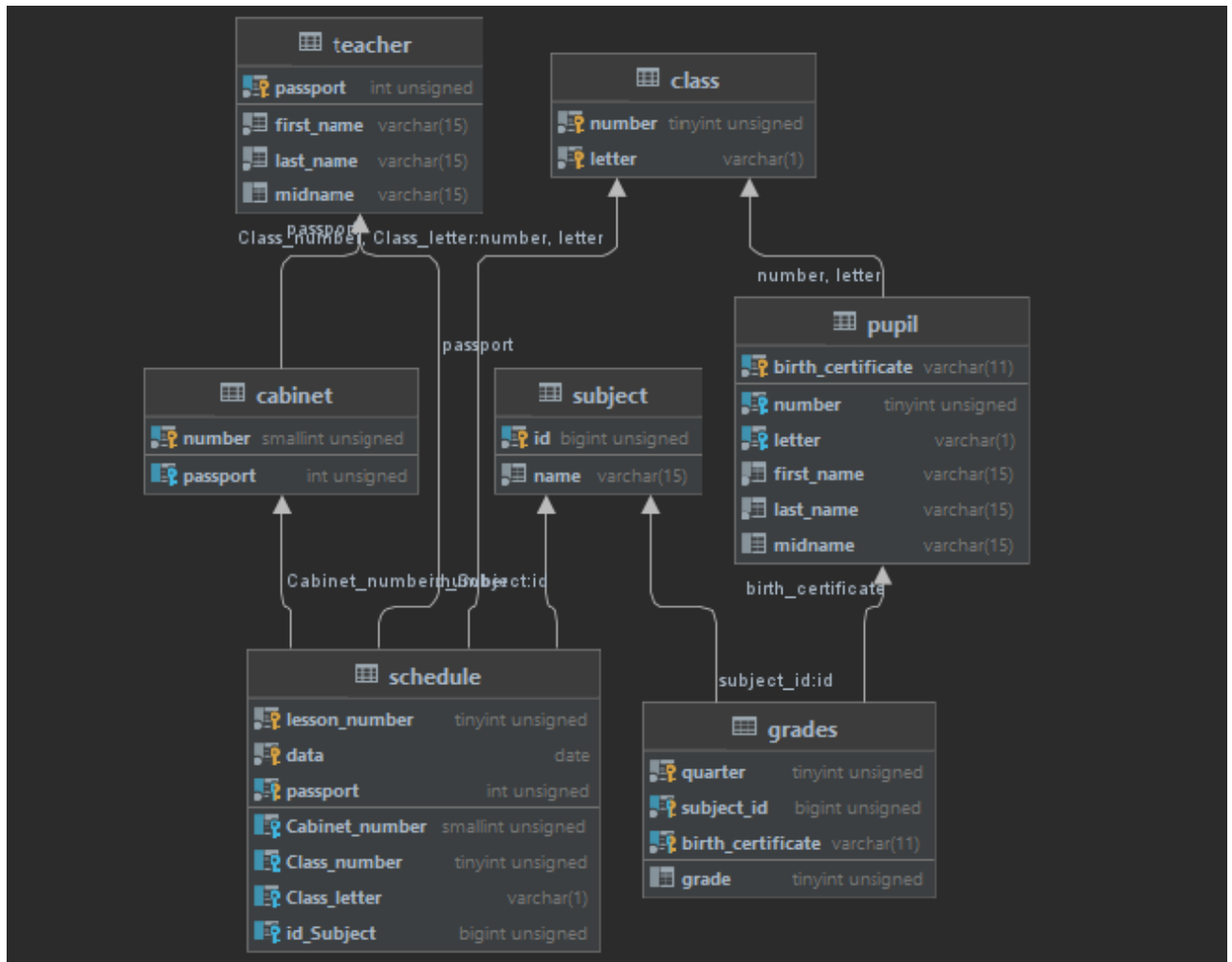


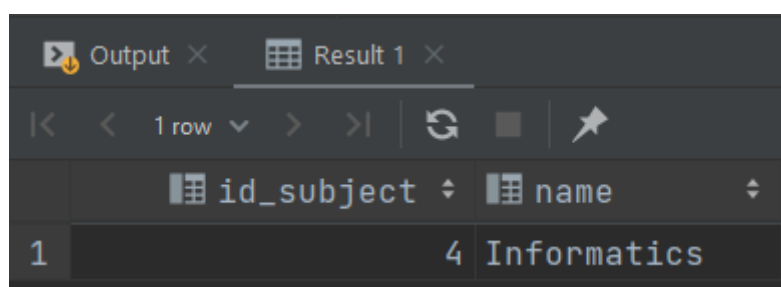
Рис. 1 – Таблицы и связи между ними в DataGrip

## Запросы, отвечающие на вопросы из ЛР1.

1. Какой предмет будет в заданном классе, в заданный день недели на заданном уроке?

```
select id_subject, name from Schedule INNER JOIN Subject ON Schedule.id_subject = Subject.  
where lesson_number = 1 and data = DATE'2021-11-15' and cabinet_number = 33;
```

Рис. 1 – Запрос (1)



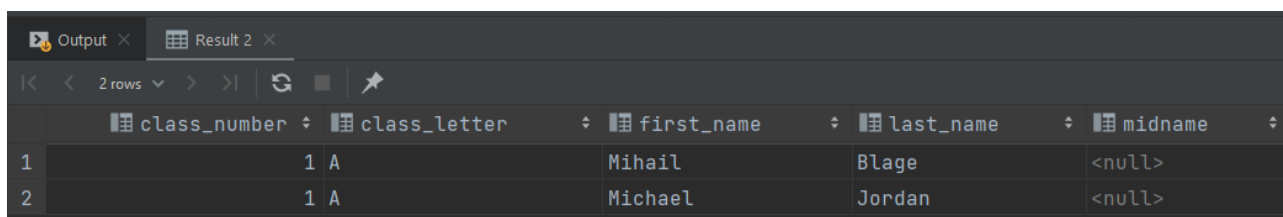
	id_subject	name
1	4	Informatics

Рис. 2 – Результат запроса (1)

2. Кто из учителей преподает в заданном классе?

```
select distinct class_number, class_letter, first_name, last_name, midname from Schedule  
INNER JOIN Teacher ON Schedule.passport = Teacher.passport  
where class_number = 1 and class_letter = 'A';
```

Рис. 3 – Запрос (2)



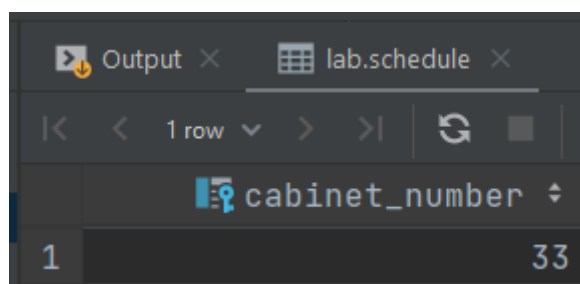
	class_number	class_letter	first_name	last_name	midname
1	1	A	Mihail	Blage	<null>
2	1	A	Michael	Jordan	<null>

Рис. 4 – Результат запроса (2)

3. В каком кабинете будет 2-й урок в среду у некоторого класса?

```
select cabinet_number from Schedule
where data = DATE'2021-11-17' and class_number = 3 and class_letter = 'B' and
lesson_number = 2;
```

*Рис. 5 – Запроса (3)*



The screenshot shows a database interface with a tab labeled 'lab.schedule'. Below the tab, there is a table with one row. The column header is 'cabinet\_number' and the value in the row is '33'.

cabinet_number
33

*Рис. 6 – Результат запроса (3)*

4. В каких классах преподает заданный предмет заданный учитель?

```
select distinct class_number, class_letter from Schedule JOIN Subject ON id_subject =
Subject.id INNER JOIN Teacher ON Schedule.passport = Teacher.passport
where name = 'Informatics' and last_name = 'Jordan'
ORDER BY class_number;
```

*Рис. 7 – Запрос (4)*

	class_number	class_letter
1	1	A
2	3	A
3	3	B
4	4	A

*Рис. 8 – Результат запроса (4)*

5. Расписание на заданный день недели для указанного класса?

```
select lesson_number as Номер_урока, name AS Название_предмета, cabinet_number AS
Кабинет, last_name AS Фамилия_учителя from Schedule JOIN Subject ON Schedule.id_subject
= Subject.id
JOIN Teacher ON Schedule.passport = Teacher.passport
where data = DATE'2021-11-18' and class_number = 3 and class_letter ='A'
ORDER BY lesson_number;
```

*Рис. 9 – Запрос (5)*

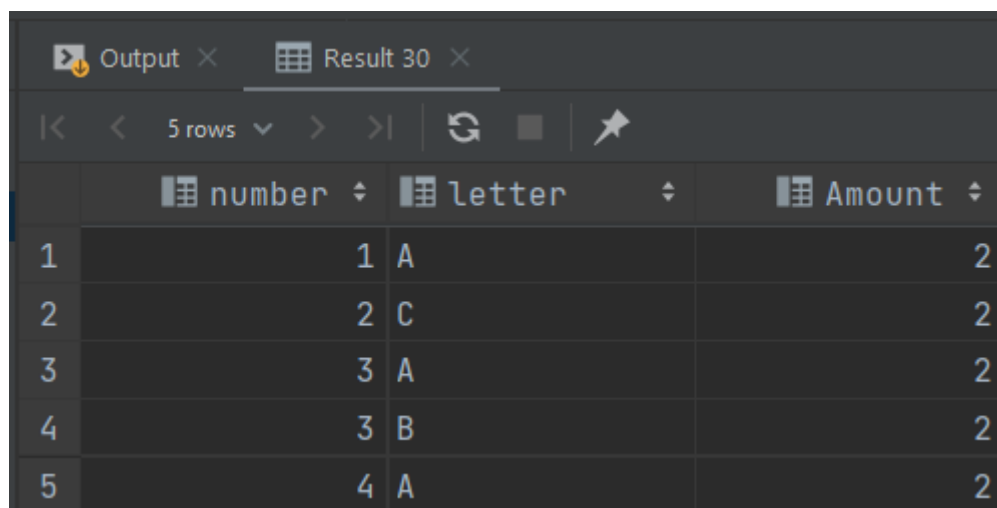
	Номер_урока	Название_предмета	Кабинет	Фамилия_учителя
1	1	Informatics	33	Jordan
2	2	Russian	1	Blage

*Рис. 10 – Результат запроса (5)*

6. Сколько учеников в указанном классе?

```
select number, letter, count(number) AS Amount from pupil  
group by number, letter;
```

Рис. 11 – Запрос (6)



The screenshot shows a database query result in a dark-themed interface. At the top, there are tabs for 'Output' and 'Result 30'. Below the tabs is a toolbar with navigation icons and a '5 rows' dropdown. The main area displays a table with the following data:

	number	letter	Amount
1	1	A	2
2	2	C	2
3	3	A	2
4	3	B	2
5	4	A	2

Рис. 12 – Результат запроса (6)

### **Выводы.**

Была построена БД в MySQL и написаны запросы, отвечающие на вопросы из ЛР1.

Если у нас есть FK, который ссылается на PK, то указывать UNIQUE и AUTO\_INCREMENT для FK (если такие свойства есть у PK) не надо.



## **Приложение.**

### **Приложение А.**

#### **DB-FIDDLE**

<https://www.db-fiddle.com/f/29LDvU2uxF3MGHTvbDHgv/6>

## **Приложение Б.**

### **PR-URL**

<https://github.com/moevm/sql-2021-9383/pull/21>

## **Приложение В.**

### **SQL-CODE**

create table Class

```
(  
    number tinyint unsigned not null,  
    letter varchar(1) not null,  
    primary key (number, letter)  
);
```

create table Pupil

```
(  
    birth_certificate varchar(11) not null,  
    number tinyint unsigned not null,  
    letter varchar(1) not null,  
    first_name varchar(15) not null,  
    last_name varchar(15) not null,  
    midname varchar(15) null,  
    constraint birth_certificate  
        unique (birth_certificate),  
    constraint Pupil_ibfk_1  
        foreign key (number, letter) references Class (number, letter),  
    constraint primary key(birth_certificate)  
);
```

create index number

on Pupil (number, letter);

create table Subject

```
(  
    id serial,
```

```

        name varchar(15) not null,
        constraint primary key(id)
    );

create table Grades
(
    quarter tinyint unsigned not null,
    subject_id bigint unsigned,
    birth_certificate varchar(11) not null,
    grade tinyint unsigned null,
    primary key (quarter, subject_id, birth_certificate),
    constraint Grades_ibfk_1
        foreign key (subject_id) references Subject (id),
    constraint Grades_ibfk_2
        foreign key (birth_certificate) references Pupil (birth_certificate)
);

create index birth_certificate
    on Grades (birth_certificate);

create table Teacher
(
    passport int unsigned not null,
    first_name varchar(15) not null,
    last_name varchar(15) not null,
    midname varchar(15) null,
    constraint passport
        unique (passport),
    constraint primary key(passport)
);

```

create table Cabinet

```
(  
    number smallint unsigned not null,  
    passport int unsigned null,  
    constraint number  
        unique (number),  
    constraint Cabinet_ibfk_1  
        foreign key (passport) references Teacher (passport),  
    constraint primary key(number)  
);
```

create table Schedule

```
(  
    lesson_number tinyint unsigned not null,  
    data date not null,  
    passport int unsigned not null,  
    cabinet_number smallint unsigned null,  
    class_number tinyint unsigned null,  
    class_letter varchar(1) null,  
    id_subject bigint unsigned not null,  
    primary key (lesson_number, data, passport),  
    constraint Schedule_ibfk_1  
        foreign key (passport) references Teacher (passport),  
    constraint Schedule_ibfk_2  
        foreign key (Cabinet_number) references Cabinet (number),  
    constraint Schedule_ibfk_3  
        foreign key (Class_number, Class_letter) references Class (number,  
letter),  
    constraint Schedule_ibfk_4
```

```
foreign key (id_Subject) references Subject (id)
);
```

```
create index Class_number
on Schedule (Class_number, Class_letter);
```

```
insert into Class values(1, 'A');
insert into Class values(2, 'C');
insert into Class values(3, 'A');
insert into Class values(3, 'B');
insert into Class values(4, 'A');
```

```
insert into Pupil values('IV-MU090200', 1, 'A', 'Dmitry', 'Gordon',
'Alexandrovich');
```

```
insert into Pupil values('I-ZI190200', 1, 'A', 'Egor', 'Lisauskas', 'Pavlovich');
```

```
insert into Pupil values('II-DA290200', 2, 'C', 'Pavel', 'Pikalov', 'Fedorovich');
```

```
insert into Pupil values('IV-NE390200', 2, 'C', 'Andrey', 'Belyakov',
'Alexandrovich');
```

```
insert into Pupil values('II-SI190200', 3, 'A', 'Pjotr', 'Shapovalov', null);
```

```
insert into Pupil values('II-NO190200', 3, 'A', 'Alexey', 'Tarannikov', null);
```

```
insert into Pupil values('I-TE190170', 3, 'B', 'Artem', 'Stepanov', null);
```

```
insert into Pupil values('I-MI190170', 3, 'B', 'Artem', 'Lukyanov', null);
```

```
insert into Pupil values('IV-KS190200', 4, 'A', 'Olesya', 'Malinina', null);
```

```
insert into Pupil values('IX-ZG190200', 4, 'A', 'Vadim', 'Efimov', null);
```

```
insert into Subject values(default, 'Physics');
```

```
insert into Subject values(default, 'Russian');
```

```
insert into Subject values(default, 'Math');
```

```
insert into Subject values(default, 'Informatics');
```

```
insert into Subject values(default, 'English');
```

insert into Teacher values(1000000000, 'Mihail', 'Blage', null);  
insert into Teacher values(2000000000, 'Derevo', 'Koren', null);  
insert into Teacher values(3000000000, 'Zolotoy', 'Gar', null);  
insert into Teacher values(4000000000, 'Pjotr', 'Junior', null);  
insert into Teacher values(4000000001, 'Michael', 'Jordan', null);

insert into Cabinet values(33, 4000000001);  
insert into Cabinet values(314, 2000000000);  
insert into Cabinet values(3, 3000000000);  
insert into Cabinet values(1, 1000000000);  
insert into Cabinet values(25, 4000000000);

insert into Grades values(1, 2, 'TV-MU090200', 2);  
insert into Grades values(1, 2, 'I-ZI190200', 3);  
insert into Grades values(1, 5, 'II-DA290200', 3);  
insert into Grades values(1, 5, 'TV-NE390200', 3);  
insert into Grades values(1, 4, 'II-SI190200', 4);  
insert into Grades values(1, 4, 'II-NO190200', 5);  
insert into Grades values(1, 1, 'I-TE190170', 5);  
insert into Grades values(1, 1, 'I-MI190170', 5);  
insert into Grades values(1, 4, 'TV-KS190200', 5);  
insert into Grades values(1, 5, 'IX-ZG190200', 5);

insert into Schedule values(1, DATE '2021-11-15', 1000000000, 1, 1, 'A', 2);  
insert into Schedule values(1, DATE '2021-11-15', 4000000001, 33, 4, 'A', 4);  
insert into Schedule values(1, DATE '2021-11-16', 1000000000, 1, 4, 'A', 2);  
insert into Schedule values(2, DATE '2021-11-16', 4000000001, 33, 1, 'A', 4);  
insert into Schedule values(1, DATE '2021-11-17', 4000000000, 25, 2, 'C', 5);  
insert into Schedule values(2, DATE '2021-11-17', 4000000001, 33, 3, 'B', 4);

```
insert into Schedule values(1, DATE '2021-11-18', 4000000001, 33, 3, 'A', 4);
insert into Schedule values(2, DATE '2021-11-18', 1000000000, 1, 3, 'A', 2);
insert into Schedule values(1, DATE '2021-11-19', 2000000000, 314, 2, 'C', 3);
insert into Schedule values(2, DATE '2021-11-19', 3000000000, 3, 3, 'B', 1);
```

```
select id_subject, name from Schedule INNER JOIN Subject ON
Schedule.id_subject = Subject.id
where lesson_number = 1 and data = DATE'2021-11-15' and cabinet_number =
33;
```

```
select distinct class_number, class_letter, first_name, last_name, midname from
Schedule INNER JOIN Teacher ON Schedule.passport = Teacher.passport
where class_number = 1 and class_letter = 'A';
```

```
select cabinet_number from Schedule
where data = DATE'2021-11-17' and class_number = 3 and class_letter = 'B' and
lesson_number = 2;
```

```
select distinct class_number, class_letter from Schedule JOIN Subject ON
id_subject = Subject.id INNER JOIN Teacher ON Schedule.passport =
Teacher.passport
where name = 'Informatics' and last_name = 'Jordan'
ORDER BY class_number;
```

```
select lesson_number as Номер_урока, name AS Название_предмета,
cabinet_number AS Кабинет, last_name AS Фамилия_учителя from Schedule JOIN
Subject ON Schedule.id_subject = Subject.id
JOIN Teacher ON Schedule.passport = Teacher.passport
where data = DATE'2021-11-18' and class_number = 3 and class_letter = 'A'
ORDER BY lesson_number;
```



```
select number, letter, count(number) AS Amount from Pupil  
group by number, letter;
```