

# Practical Lab 1

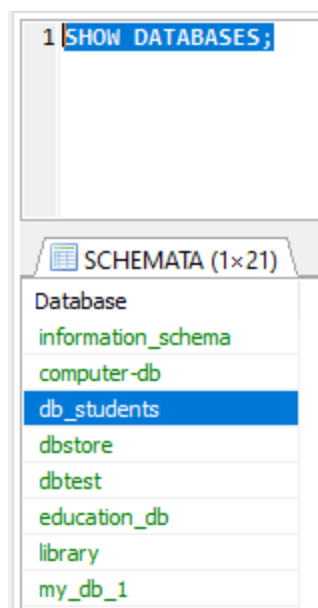
1/ Create the database for students:

```
CREATE DATABASE DB_students;
```

```
CREATE DATABASE DB_students; //DB students  
CREATE DB DB_students; //DATABASE
```

2/ Displays all databases stored in the server

```
SHOW DATABASES;
```



3/ In the future, we will create tables inside DB\_students, so we to ...

```
USE DB_students;
```

4/ Create new table for departments

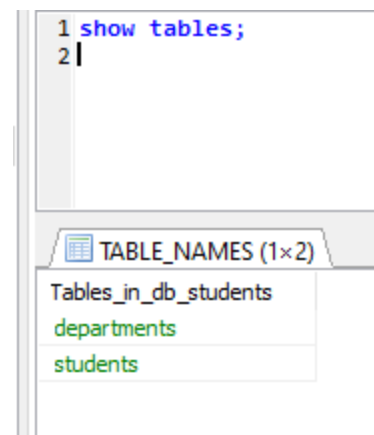
```
create table departments (  
    id_dept int primary key,  
    name_dept varchar (50)  
);
```

5/ create new table for students

```
create table students (  
    id_student int primary key,  
    name varchar(50),  
    age int,  
    address varchar(100),  
    city varchar(30) default 'Afif',  
    id_department int  
);
```

6/Display all tables in DB\_students

```
show tables;
```

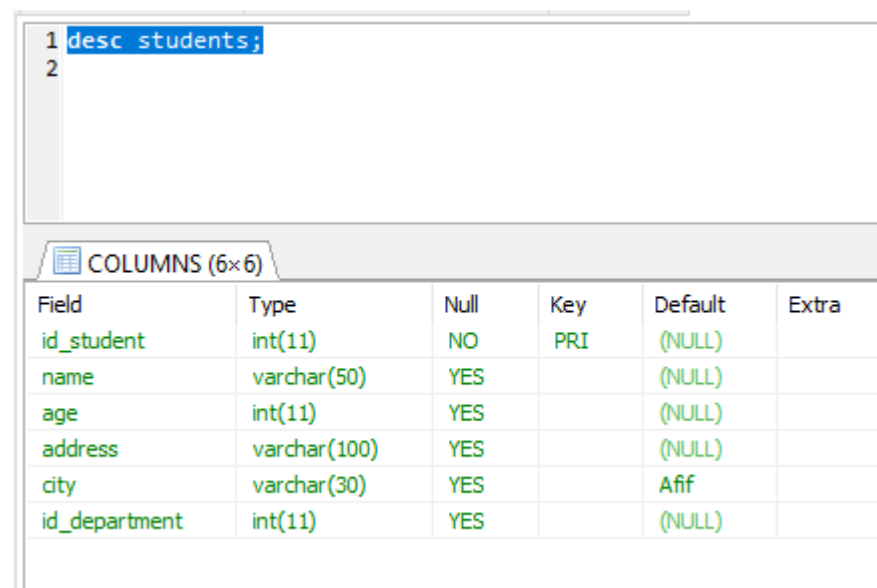


The screenshot shows a database interface with a command prompt area at the top containing the text '1 show tables;' and '2 |'. Below the prompt is a table titled 'TABLE\_NAMES (1x2)' which lists the tables in the database: 'Tables\_in\_db\_students', 'departments', and 'students'.

Tables_in_db_students
departments
students

7/ Describe table students

```
desc students;
```



The screenshot shows a database interface with a command prompt area at the top containing the text '1 desc students;' and '2'. Below the prompt is a table titled 'COLUMNS (6x6)' which displays the structure of the 'students' table. The table has six columns: Field, Type, Null, Key, Default, and Extra. The rows represent the fields: id\_student, name, age, address, city, and id\_department.

Field	Type	Null	Key	Default	Extra
id_student	int(11)	NO	PRI	(NULL)	
name	varchar(50)	YES		(NULL)	
age	int(11)	YES		(NULL)	
address	varchar(100)	YES		(NULL)	
city	varchar(30)	YES		Afif	
id_department	int(11)	YES		(NULL)	

8/ add computer science department in the table department

```
insert into departments (id_dept, name_dept) values (1, 'Computer Science');
```

```
insert into departments (id_dept, name_dept) values ('Computer Science', 2);
```

```
insert into departments (id_dept, name_dept) values (1, 'Computer Science'); // 1 primary key
```

```
insert into departments (id_dept, name_dept) values (2, 'Math', 'mathematical department');
```

9/ add Math and English department in the same SQL insert query

⇒ Solution: multiple insert

```
insert into departments (id_dept, name_dept)
values (2, 'Math'), (3, 'English');
```

10/ Prb:

```
insert into departments (name_dept)
values ('Science');
```

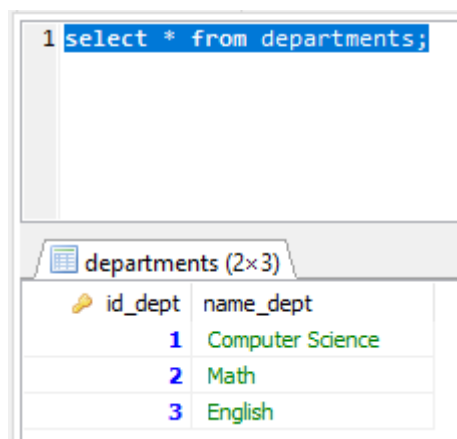
⇒ Solution: auto\_increment

```
create table departments (
    id_dept int primary key auto_increment,
    name_dept varchar (50)
);
```

```
create table students (
    id_student int primary key auto_increment,
    name varchar(50),
    age int,
    address varchar(100),
    city varchar(30) default 'Afif',
    id_department int
);
```

11/ Display the content of table department

```
select * from departments;
```



The screenshot shows a database management tool interface. At the top, a text area contains the SQL query: `1 select * from departments;`. Below this, a tab labeled 'departments (2x3)' is active. Underneath the tab is a table with two columns: 'id\_dept' (marked with a primary key icon) and 'name\_dept'. The table contains three rows of data.

id_dept	name_dept
1	Computer Science
2	Math
3	English

11/ Remove the English department

```
delete from departments where id_dept = 3;
```

12/ Modify the name of Computer Science to Computer

```
update departments set name_dept = 'Computer' where id_dept = 1;
```

```
update table departments set name_dept = 'Computer' where id_dept =  
1;
```

13/ delete all the content of the table department

```
delete from departments;
```