|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Fields | Datatypes | Null | Key | Default | Check | extra |
| student\_id | Int(11) | No | Primary |  |  | Auto\_Increment |
| name | varchar(50) | N0 |  |  |  | Unique |
| address | varchar(100) | No |  | Birtamode |  |  |
| class\_id | int(11) | No | Foreign |  |  |  |
| section | varchar(50) | Yes |  |  |  |  |
| age | int(11) | No |  | 16 | age>=15 |  |

1. Write SQL Query to create following table(Student).

**Note**: Foreign key reference to (Class) Table.

SQL query:

Database sujan:

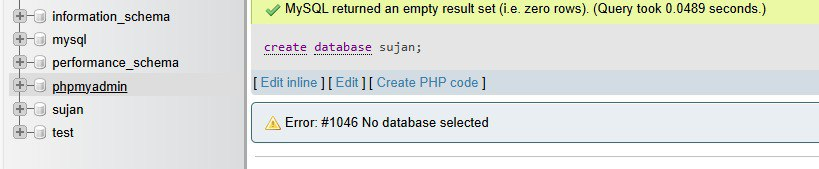


Table class:

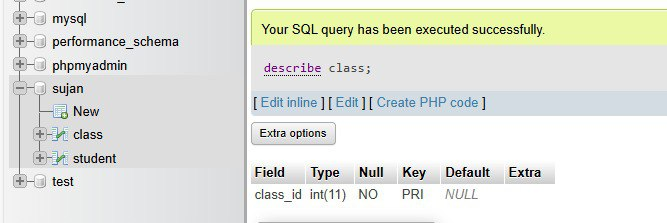
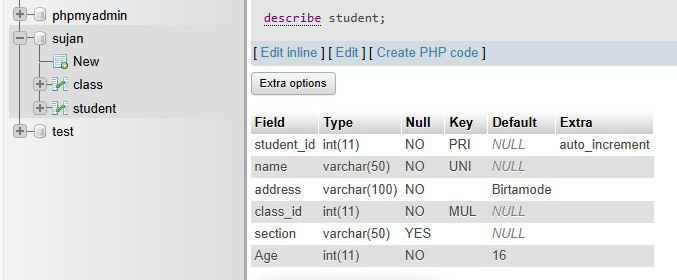
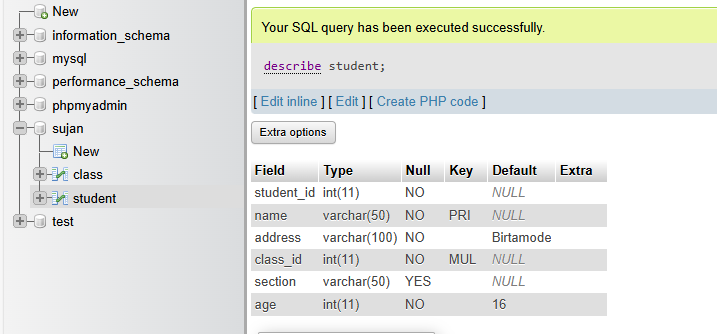


Table student:



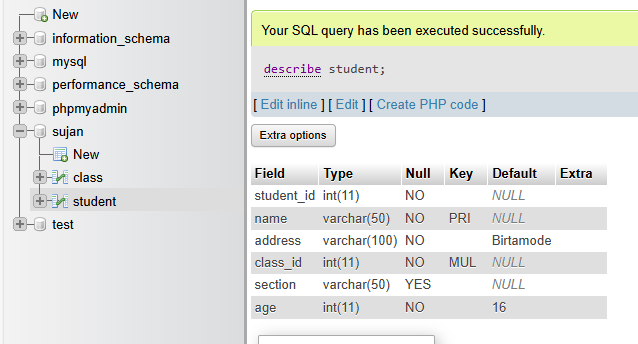
2. Write SQL query to drop primary key from above table.

SQL query:



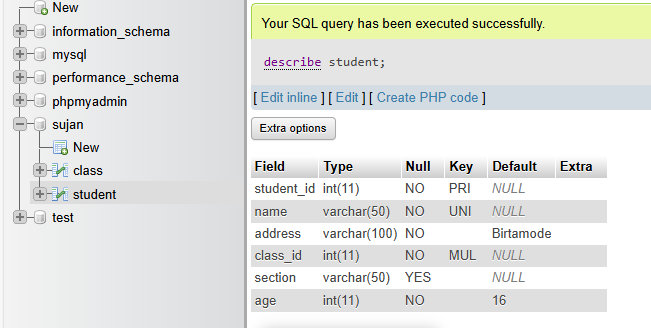
3. Write SQL query to drop foreign key from above table.

SQL query:



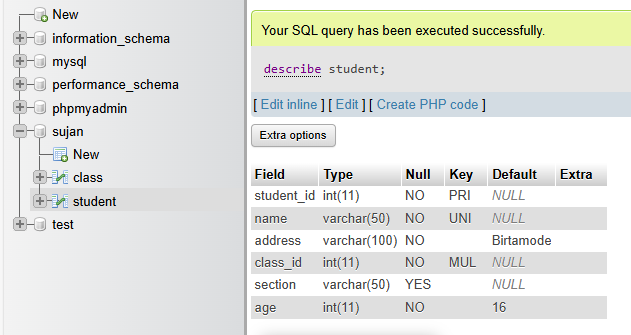
4. Write SQL query to set student id as primary key.

SQL query:



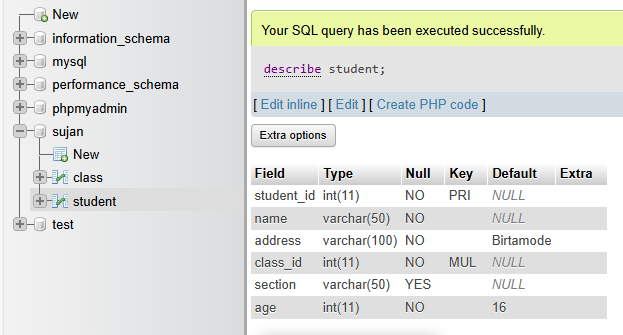
5. Write SQL query to set class id as foreign key.

SQL query:



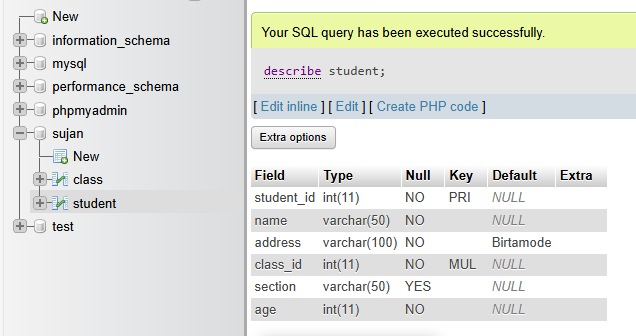
6. Write SQL query to remove unique constraint from name.

SQL query:



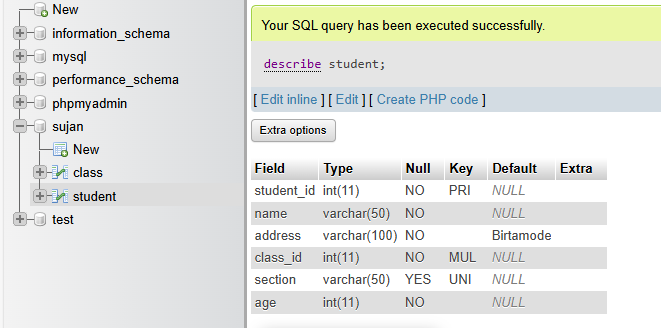
7. Write SQL query to remove default constraint from age.

SQL query:



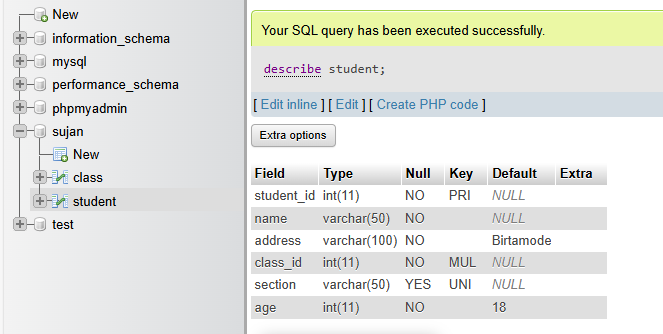
8. Write SQL query to add unique constraint to section.

SQL query:



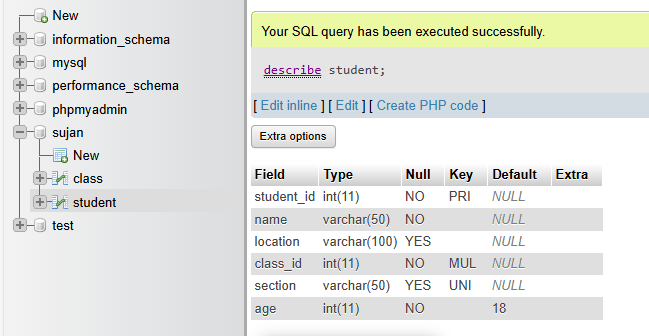
9. Write SQL query to add default value 18 to age.

SQL query:



10. Write SQL query to change column name address to location.

SQL query:

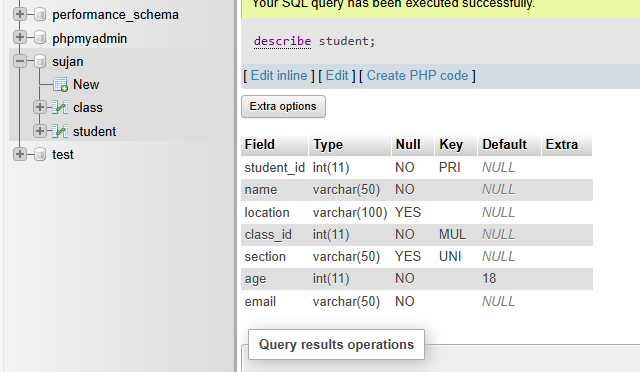


11. Write SQL query to add new column email and make it not null.

SQL query:

alter table student

add email varchar(50) not null;

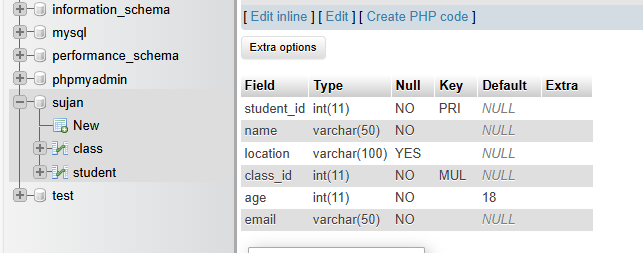


12. Write SQL query to remove column section from above table.

SQL query:

alter table student

drop column section;

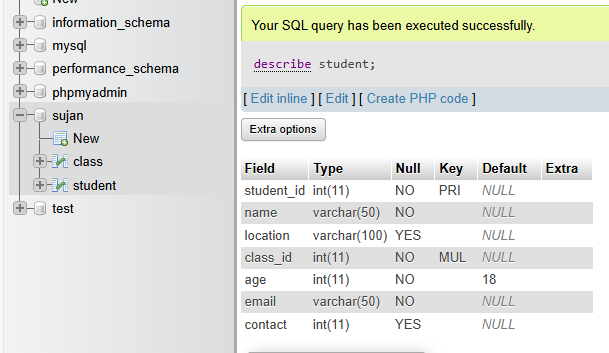


13. Write SQL query to add new column contact and make data type as integer.

SQL query:

[alter](http://localhost:8084/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/alter-table.html) [table](http://localhost:8084/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/alter-table.html) student

add contact [int](http://localhost:8084/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/numeric-types.html)

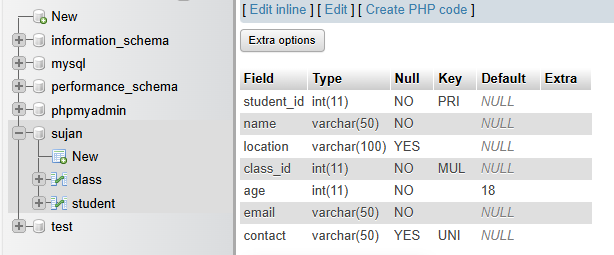


14. Write SQL query to change data type of column contact to varchar and make it unique.

SQL query:

[alter](http://localhost:8084/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/alter-table.html) [table](http://localhost:8084/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/alter-table.html) student

modify contact [varchar](http://localhost:8084/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/string-types.html)(50) unique

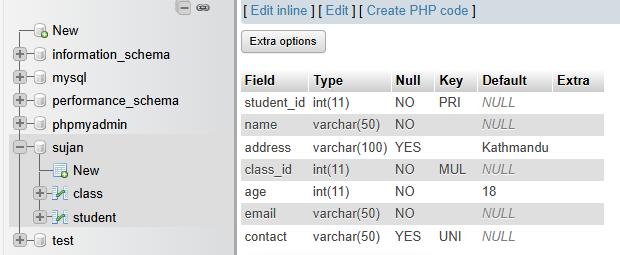


15. Write SQL query to change default value of address to Kathmandu.

SQL query:

[alter](http://localhost:8084/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/alter-table.html) [table](http://localhost:8084/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/alter-table.html) student

alter column location [set](http://localhost:8084/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/set.html) [default](http://localhost:8084/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/miscellaneous-functions.html#function_default) 'Kathmandu'



16. Insert five set of records in above table.

SQL query:

insert into class

values(1001);

insert into student

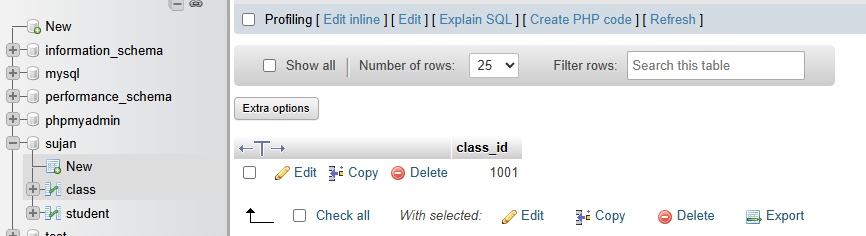
values(1,"aarya","ktm",1001,20,"aaryadhungana@gmail.com","9898989898"),

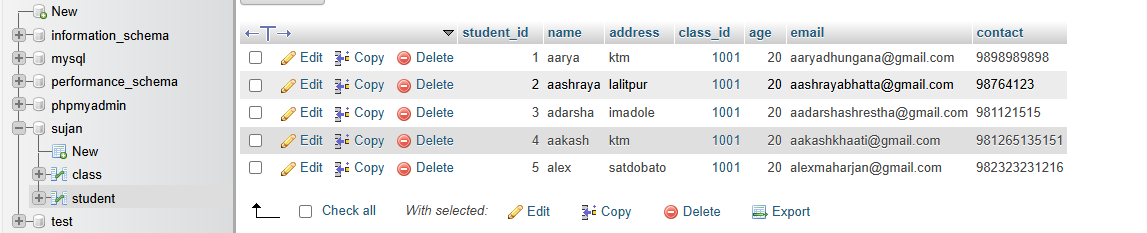
(2,"aashraya","lalitpur",1001,20,"aashrayabhatta@gmail.com","98764123"),

(3,"adarsha","imadole",1001,20,"aadarshashrestha@gmail.com","981121515"),

(4,"aakash","ktm",1001,20,"aakashkhaati@gmail.com","981265135151"),

(5,"alex","satdobato",1001,20,"alexmaharjan@gmail.com","982323231216");





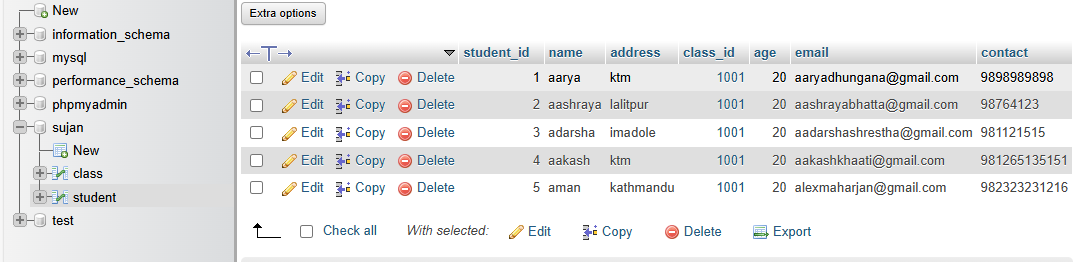
17. Write SQL query to update name and address of student whose student id is 5.

SQL query:

update student

set name="aman", address="kathmandu"

where student\_id=5;

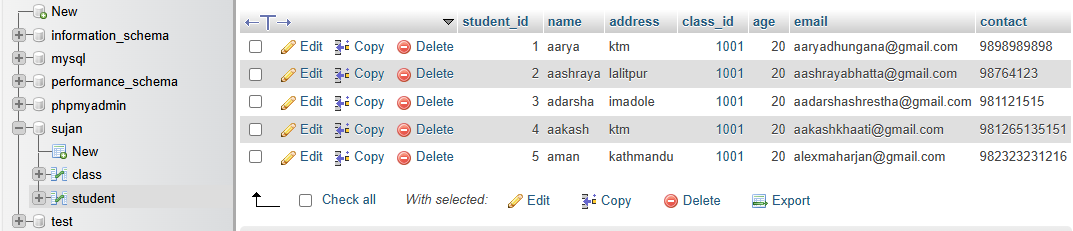


18. Write SQL query to delete all the records of student having age greater than 20.

SQL query:

delete from student

where age>20;



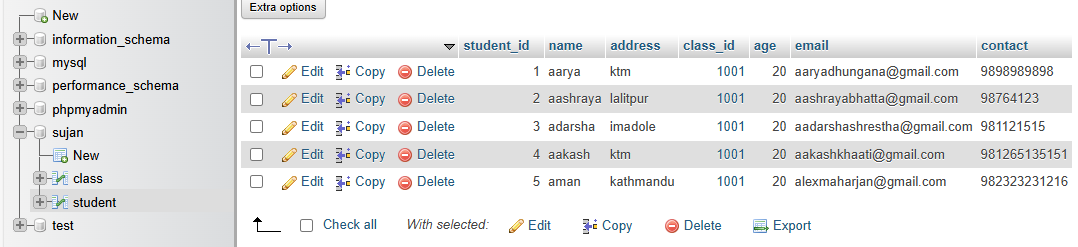
19. Write SQL query to update age of student having address btm.

SQL query:

update student

set age=21

where address=”btm”;

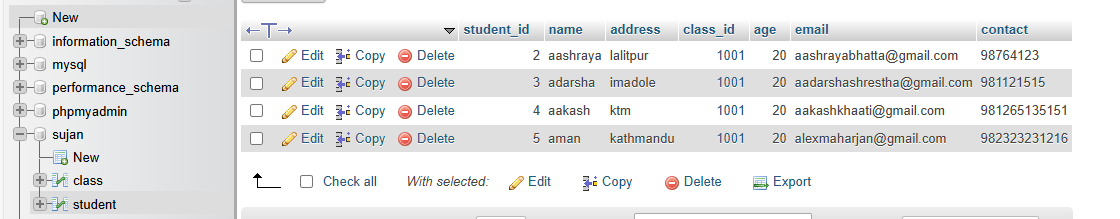


20. Write SQL query to delete all records of student having student id 1.

SQL query:

delete from student

where student\_id=1;



alter table student

add section varchar(50)

insert into class

values(1001),

(5),

(6),

(4)

insert into student

values(1,"Aarya","ktm",1001,20,"aaryadhungana@gmail.com","9898989898","A"),

(2,"Aashraya","lalitpur",1001,20,"aashrayabhatta@gmail.com","98764123","A"),

(3,"Adarsha","imadole",1001,20,"aadarshashrestha@gmail.com","981121515","A"),

(4,"Aakash","ktm",1001,20,"aakashkhaati@gmail.com","981265135151","A"),

(5,"Alex","satdobato",1001,20,"alexmaharjan@gmail.com","982323231216","B"),

(6,"John","Birtamode",5,23,"johnhopkins@gmail.com","9808768112" ,"B"),

(7,"Rahul","Bhaktapur",5,24,"rahul@gmail.com","98087112" ,"B"),

(8,"Sabin","Jawalakhel",5,22,"sabinchettri@gmail.com","981268155" ,"B"),

(9,"sahil","nakhipot",4,21,"sahilkarki@gmail.com","98423423455" ,"B"),

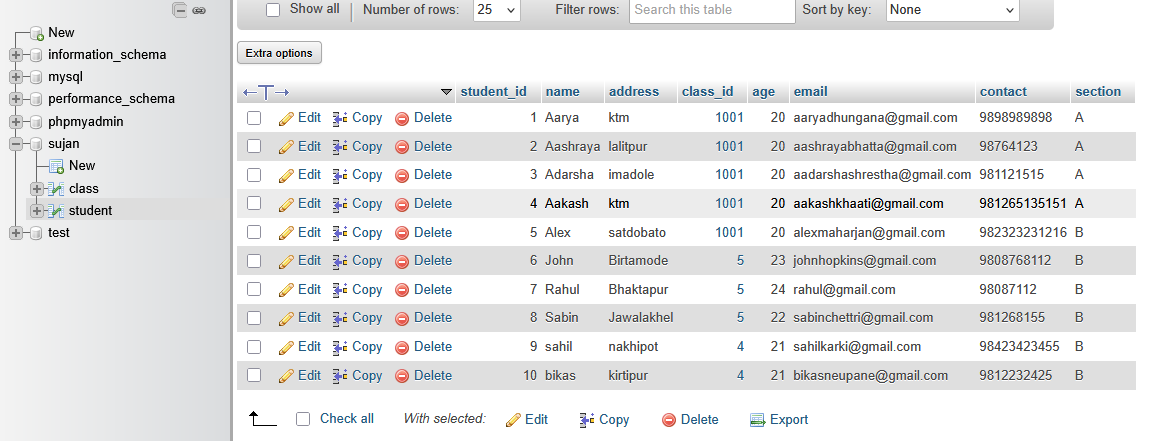
(10,"bikas","kirtipur",4,21,"bikasneupane@gmail.com","9812232425" ,"B")

21. Write SQL query to select all records of student.

SQL query:

Select \* from student

where 1;



22. Write SQL query to select all records of student having student id 3.

SQL query:

Select \* from student

Where student\_id=3;

A screenshot of a computer

AI-generated content may be incorrect.

23. Write SQL query to select name and address of students whose age is greater than 21.

SQL query:

Select name, address

From student

Where age>21;

A screenshot of a computer

AI-generated content may be incorrect.

24. Write SQL query to select student id and name of students whose address in Birtamode.

SQL query:

Select student\_id, name

From student

Where address= 'Birtamode ';

A screenshot of a computer

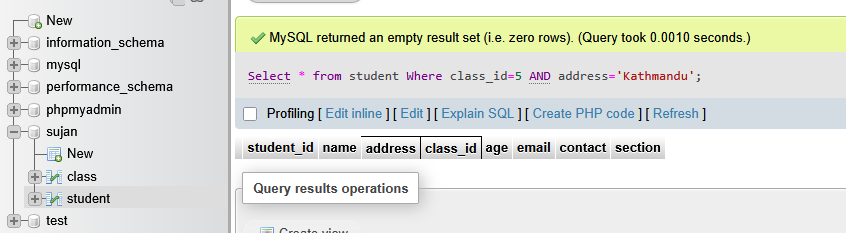
AI-generated content may be incorrect.

25. Write SQL query to select records of students whose class id is 5 and address is Kathmandu.

SQL query:

Select \* from student

Where class\_id=5 AND address=’Kathmandu’



26. Write SQL query to select maximum age from above table.

SQL query:

Select max(age) as max\_age

FROM student;

A screenshot of a computer

AI-generated content may be incorrect.

27. Write SQL query to select minimum age of students whose address is Birtamode.

SQL query:

Select min(age) as min\_age

From student

Where address= 'Birtamode ';

A screenshot of a computer

AI-generated content may be incorrect.

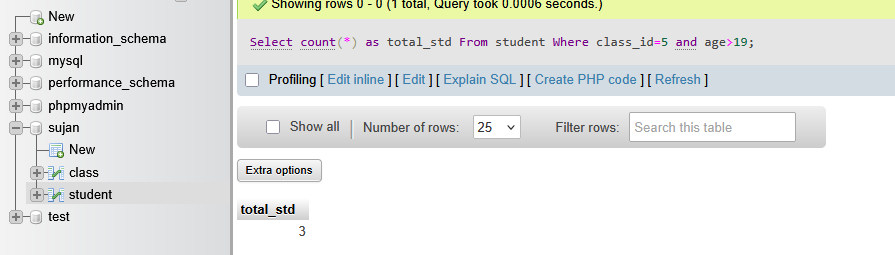
28. Write SQL query to find total number of students having class id 5 and age greater than 19.

SQL query:

Select count(\*) as total\_std

From student

Where class\_id=5 and age>19;



29. Write SQL query to find average age of students whose class id is 4 and section is B.

SQL query:

Select avg(age) as avg\_std

From student

Where class\_id=4 and section= 'B ';

A screenshot of a computer

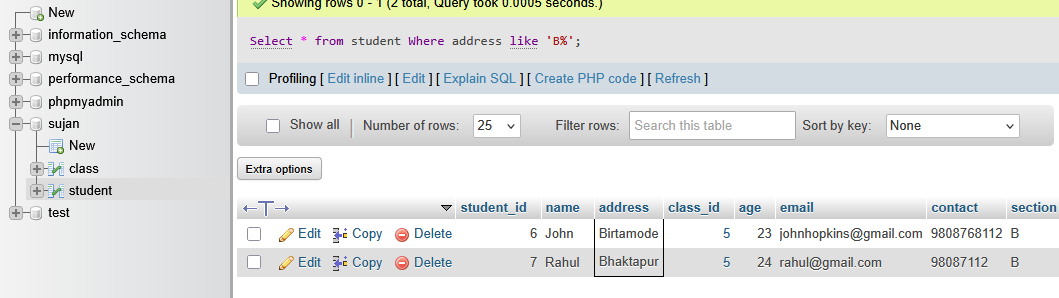
AI-generated content may be incorrect.

30. Write SQL query to select students whose address starts with letter ‘B’.

SQL query:

Select \* from student

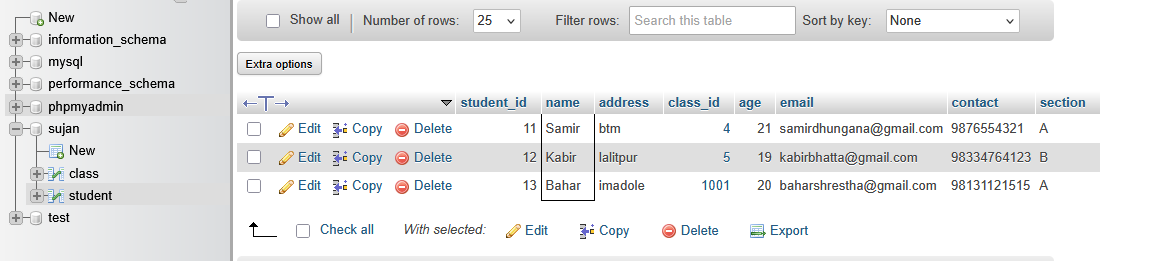
Where address like 'B% ';



31. Write SQL query to count those students whose name ends with letter ‘R’.

Select \* from student

Where name like '%R';

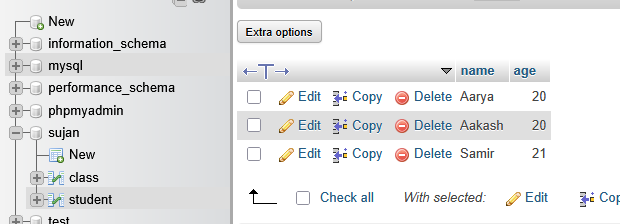


32. Write SQL query to select name and age of students whose having address btm or ktm.

Select name, age

From student

Where address= 'btm' or address=’ktm’;



33. Write SQL query to select sum of age of students having id 1,2 and 3.

Select sum(age) as sum\_age

From student

Where student\_id between 1 and 3;

A screenshot of a computer

AI-generated content may be incorrect.

34. Write SQL query to select students whose age is between 18 and 22.

Select \*

From student

Where age between 18 and 22;

A screenshot of a computer

AI-generated content may be incorrect.

35. Write SQL query to select total students of each age group.

SELECT age, COUNT(\*) AS total\_students

FROM student

GROUP BY age;

A screenshot of a computer

AI-generated content may be incorrect.

36. Write SQL query to select class id, name and maximum age of students studying in each class.

SELECT class\_id, max(age) as max\_age

FROM student

GROUP BY class\_id;

A screenshot of a computer

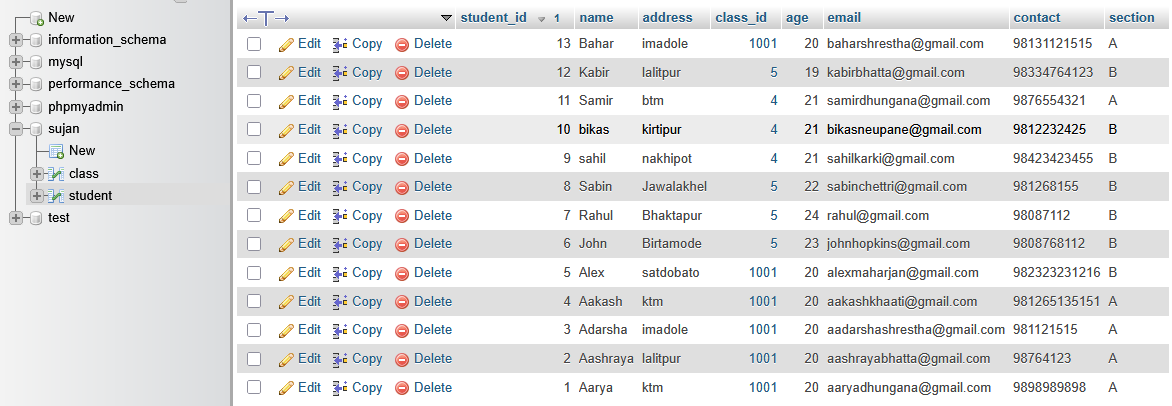
AI-generated content may be incorrect.

37. Write SQL query to select student’s records by arranging in descending order on the basis of student id.

SELECT \*

FROM student

ORDER BY student\_id DESC;



38. Write SQL query to select student id and name by of students whose age is greater than 20 after arranging records in alphabetical order on the basis of name.

SELECT student\_id, name

FROM student

WHERE age > 20

ORDER BY name ASC;

A screenshot of a computer

AI-generated content may be incorrect.

39. Write SQL query to select records of student whose age is maximum among all the students.

SELECT \*

FROM student

WHERE age = (SELECT MAX(age) FROM student);

A screenshot of a computer

AI-generated content may be incorrect.

40. Write SQL query to select student id and name of student whose student id is maximum among all the students.

SELECT student\_id, name

FROM student

WHERE student\_id = (SELECT MAX(student\_id) FROM student);

A screenshot of a computer

AI-generated content may be incorrect.

41. Write SQL query to select name and age of student whose age is minimum than the

average age of all students.

SQL query:

select name, age

from student

where age < (select avg(age) from student);

A screenshot of a computer

AI-generated content may be incorrect.

42. Write SQL query to list all the students except ‘btm & ‘ktm in asc order of age.

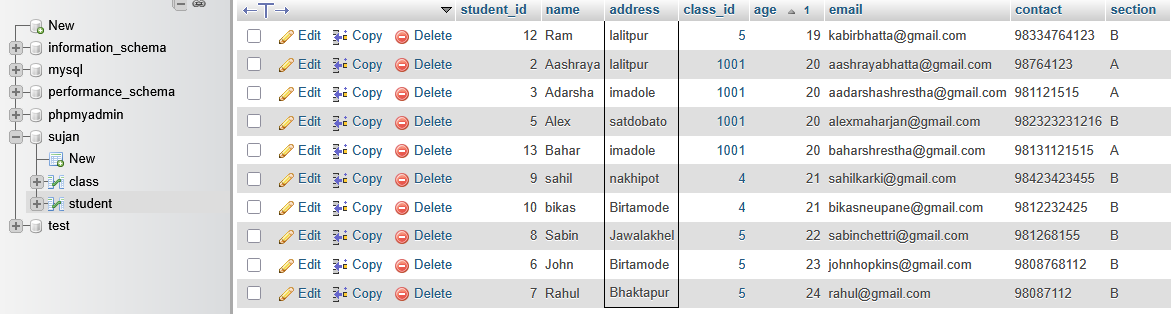
SQL query:

Select \*

From student

Where address not in ('btm','ktm')

Order by age asc;



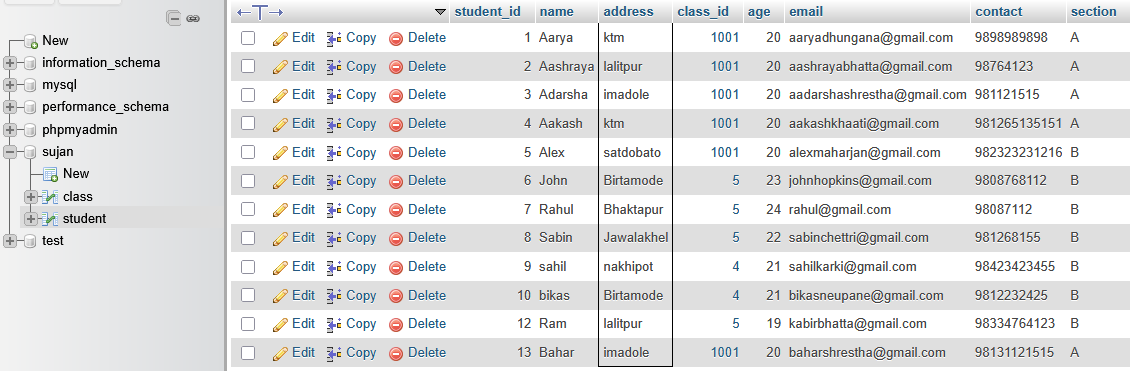
43. Write SQL query the students who does not belong to address ‘btm’.

SQL query:

Select \*

From student

Where address <> ‘btm’;



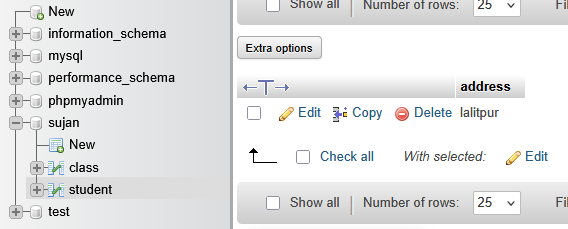
44. Write SQLL query to display the location of ‘Ram’’.

SQL query:

Select address

From student

Where name = 'Ram';



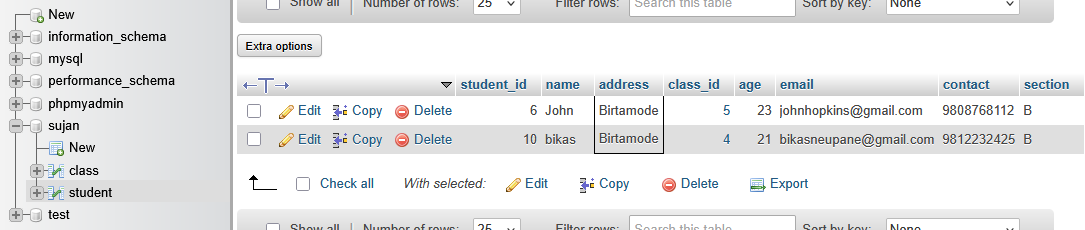
45. Write SQL query to display the total information of student table along with name and location of all the students having address ‘Birtamode’.

SQL query:

Select \*

From student

Where address = ‘birtamode’;



46. Create table below with appropriate data type and constraints.

**Employee**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Emp\_Id | Name | Address | Salary | Dept\_Id |

**Department**

|  |  |  |
| --- | --- | --- |
| Dept\_Id | Dept\_Name | Floor |

SQL query:

CREATE table department

(

dept\_id int PRIMARY KEY,

dept\_name varchar(30),

floor varchar(30)

);

create table employee

(

emp\_id int PRIMARY KEY,

name varchar(30) not null,

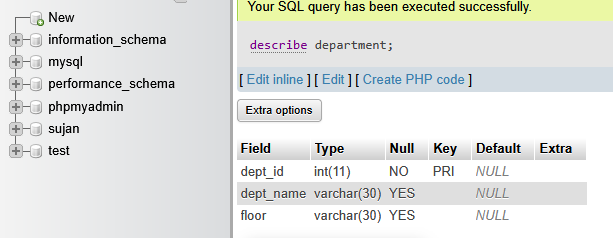
address varchar(30) not null,

salary int not null,

dept\_id int,

CONSTRAINT em\_di\_fk FOREIGN KEY (dept\_id) REFERENCES department(dept\_id)

);



A screenshot of a computer

AI-generated content may be incorrect.

47. Use all types of joins to select employee id, name and department name of employees.

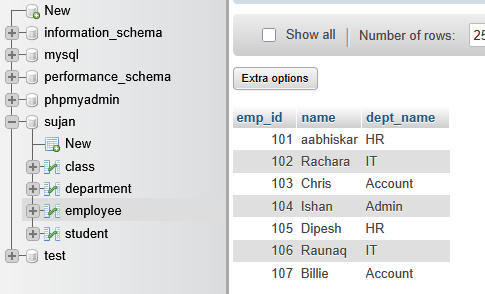
SQL query:

INNER JOIN

SELECT emp\_id, name, dept\_name

FROM employee INNER JOIN department

ON employee.dept\_id = department.dept\_id;

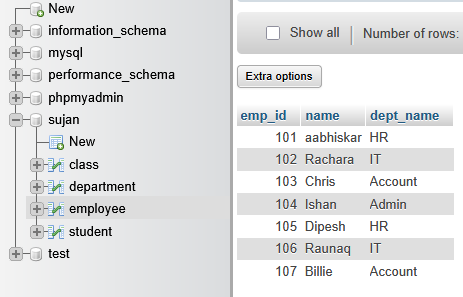


LEFT OUTER JOIN

SELECT emp\_id, name, dept\_name

FROM employee LEFT JOIN department

ON employee.dept\_id = department.dept\_id;

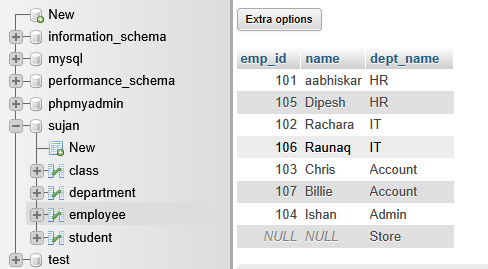


RIGHT OUTER JOIN

SELECT emp\_id, name, dept\_name

FROM employee RIGHT JOIN department

ON employee.dept\_id = department.dept\_id;



FULL JOIN

SELECT emp\_id, name, dept\_name

FROM employee

LEFT JOIN department

ON empolyee.dept\_id = department.dept\_id

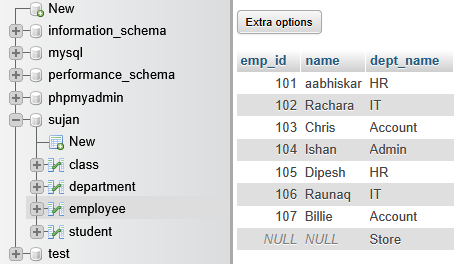
UNION

SELECT emp\_id, name, dept\_name

FROM employee

RIGHT JOIN department

ON empolyee.dept\_id = department.dept\_id;

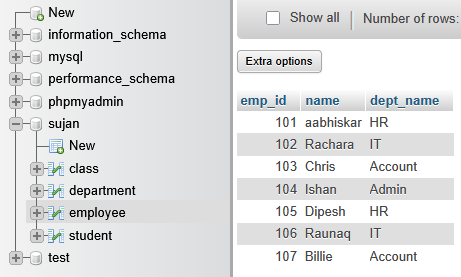


NATURAL JOIN

SELECT emp\_id, name, dept\_name

FROM employee

NATURAL JOIN department;



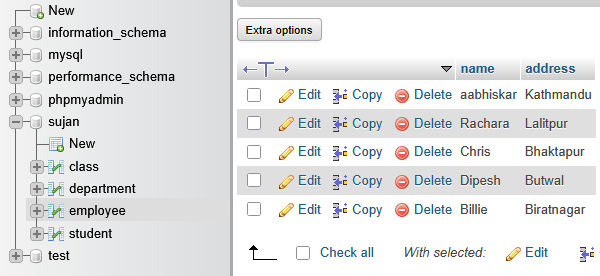
48. Select name and address of employees whose salary is between 10000 and 20000.

SQL query:

select name,address

from employee

where salary between 10000 and 20000;



49. Select employee id, employee name and department name of employees working in

first floor.

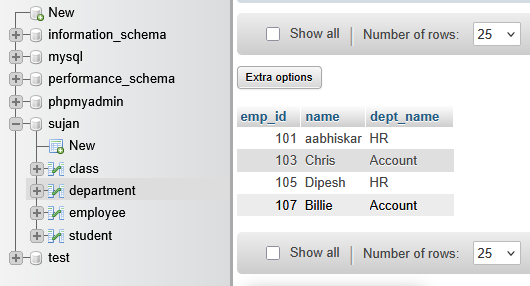
SQL query:

select emp\_id, name, dept\_name

from department

NATURAL JOIN employee

where floor =1;



50. Select all records of department which are in second floor.

SQL query:

select \* from department

where floor =2;

A screenshot of a computer

AI-generated content may be incorrect.