DBMS - Mini Project
School Management System

Submitted By: Kaushal Suresh

SRN : PES1UG20CS201

V Semester Section D

Short Description and Scope of the Project

Student data is confidential for each school and should be protected with utmost care. A database management system (DBMS) within school ERP software is a perfect tool for that.

The student and employee data help organize school-related tasks and activities. For that, organizing the data with high security is necessary. The traditional way of managing such a large database includes either using paper or manually adding each entry in spreadsheets. In each case, there is a high chance of human error and the data getting damaged due to physical factors.

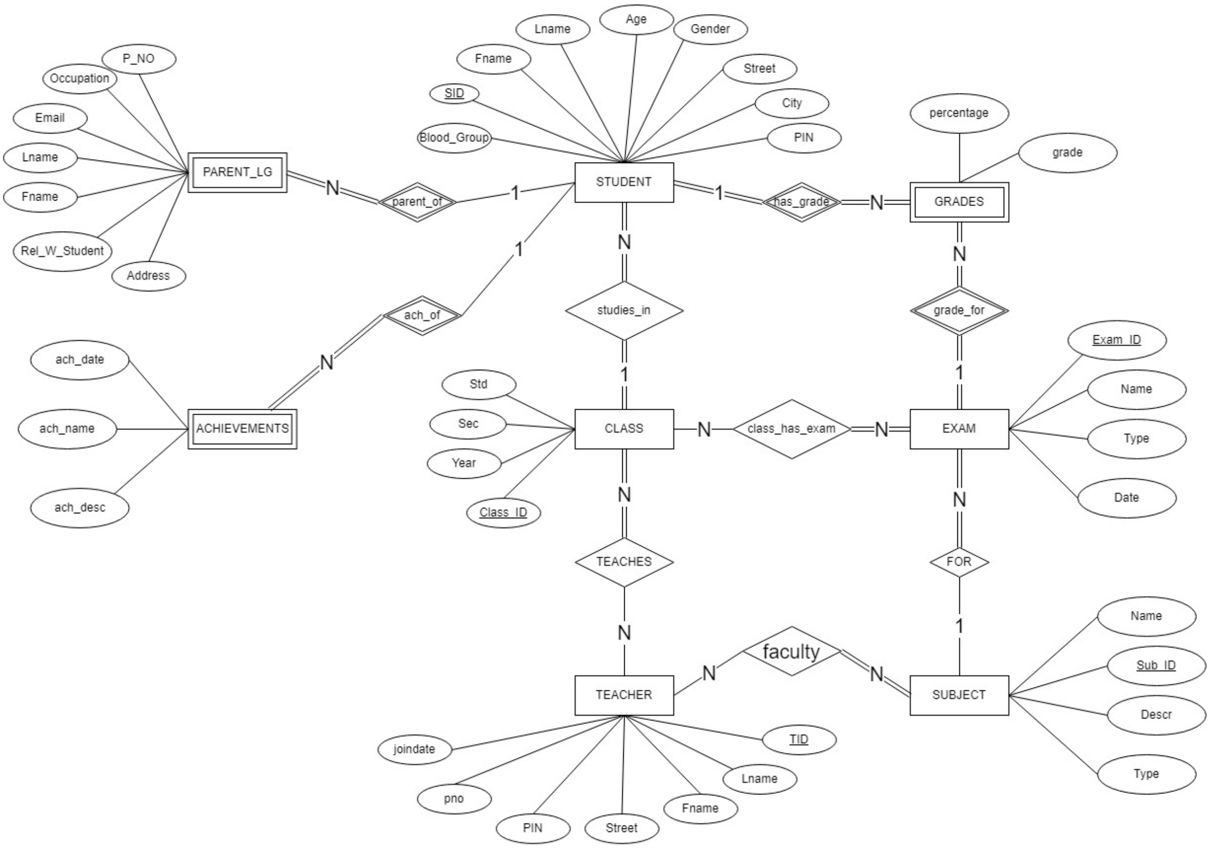
With the growing digitization, having a digital automated solution for data management is extremely important. It not only helps you manage your data without human interference but also connects with other modules of the school management software to allow better organization of different tasks and functions of your school.

The database contains the following tables and provides the following features:

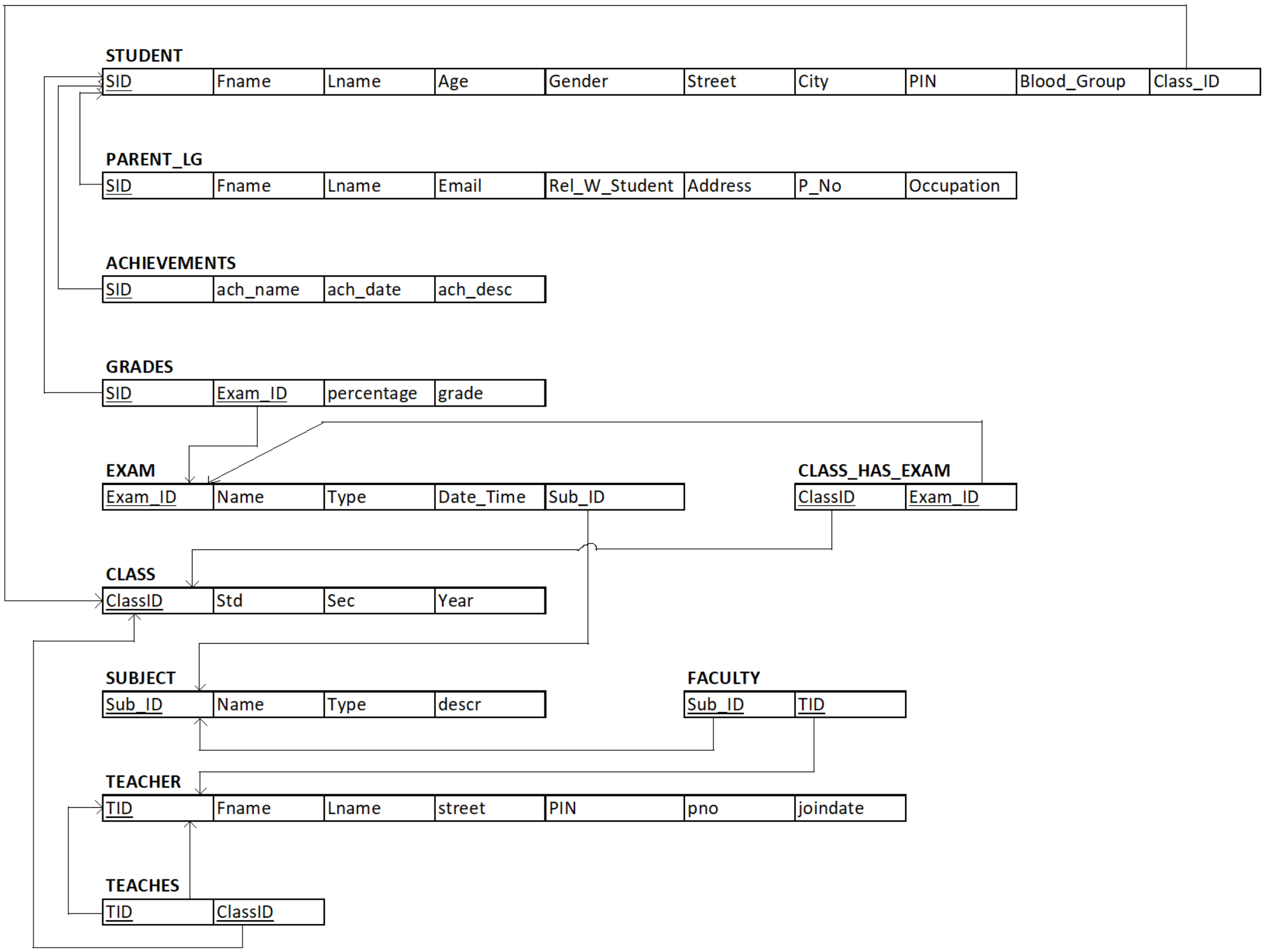
- achievements
- class
- exam
- faculty
- grades
- parent_lg
- student
- subject
- teacher

The scope of school management is very large. It includes everything regarding the efficient functioning of the educational institution, securing the greatest benefit to the greatest number through an adoption of practical measures.

ER Diagram


P_NO
Occupation
Email
Lname
Fname
Rel_W_Student
Address
PARENT_LG
N
parent_of
1
Lname
Age
Gender
Fname
Street
SID
City
Blood_Group
PIN
STUDENT
1
has_grade
N
GRADES
percentage
grade
1
ach_of
N
ACHIEVEMENTS
ach_date
ach_name
ach_desc
N
studies_in
1
N
grade_for
1
Exam_ID
Name
Type
Date
EXAM
Std
Sec
Year
Class_ID
CLASS
N
class_has_exam
N
N
TEACHES
N
N
FOR
1
TEACHER
N
faculty
N
SUBJECT
Name
Sub_ID
Descr
Type
joindate
pno
PIN
Street
Fname
Lname
TID

Relational Schema



STUDENT

SID	Fname	Lname	Age	Gender	Street	City	PIN	Blood_Group	Class_ID

PARENT_LG

SID	Fname	Lname	Email	Rel_W_Student	Address	P_No	Occupation

ACHIEVEMENTS

SID	ach_name	ach_date	ach_desc

GRADES

SID	Exam_ID	percentage	grade

EXAM

Exam_ID	Name	Type	Date_Time	Sub_ID

CLASS_HAS_EXAM

ClassID	Exam_ID

CLASS

ClassID	Std	Sec	Year

SUBJECT

Sub_ID	Name	Type	descr

FACULTY

Sub_ID	TID

TEACHER

TID	Fname	Lname	street	PIN	pno	joindate

TEACHES

TID	ClassID

DDL statements - Building the database

CREATING DATABASE:

```
create database school_management;
```

ADDING TABLES:

```
create table class (Std int(2), Sec varchar(5), Year int (5), ClassID int(5), PRIMARY
KEY(ClassID);
```

```
create table student (SID varchar(25) NOT NULL, Fname varchar(255), Lname
varchar(255),  Age int(3), Gender varchar(225) CHECK (Gender in
('M','F','m','f','Male','Female','male','female','MALE','FEMALE','Unknown','unknown','
UNKNOWN')), Street varchar(255), City varchar(255), PIN varchar(15), Blood_Group
varchar(5), ClassID int(5), PRIMARY KEY(SID), FOREIGN KEY(ClassID) REFERENCES
class(ClassID) ON UPDATE CASCADE ON DELETE SET NULL);
```

```
create table parent_LG (SID varchar(25), Fname varchar(255), Lname varchar(255),
Email varchar(255), Rel_W_Student varchar(25), Address varchar(255), P_No
varchar(15), Occupation varchar(255), PRIMARY KEY(SID, Rel_W_Student), FOREIGN
KEY (SID) REFERENCES student(SID) ON UPDATE CASCADE ON DELETE CASCADE);
```

```
CREATE TABLE `achievements` (  `SID` varchar(25),  `ach_name`
varchar(100),  `ach_date` date,  `ach_desc` varchar(255),   FOREIGN KEY (`SID`)
REFERENCES `student` (`SID`) ON UPDATE CASCADE ON DELETE CASCADE)
```

```
CREATE TABLE `teacher` (  `TID` varchar(25) NOT NULL,  `Fname`
varchar(255,  `Lname` varchar(255),  `street` varchar(255,  `PIN` varchar(10),  `pno`
varchar(15), joindate date,  PRIMARY KEY (`TID`) )
```

```
CREATE TABLE `subject` (  `Sub_ID` varchar(50) NOT NULL,  `Name`
varchar(255),  `Type` varchar(50),  `descr` varchar(255),  PRIMARY KEY (`Sub_ID`) )
```

```
CREATE TABLE `exam` (  `Exam_ID` varchar(25) NOT NULL,  `Name`
varchar(255),  `Type` varchar(255),  Date date, Time time,  `Sub_ID`
varchar(50),  PRIMARY KEY (`Exam_ID`),   FOREIGN KEY (`Sub_ID`) REFERENCES
`subject` (`Sub_ID`) ON UPDATE CASCADE ON DELETE CASCADE)
```

```
CREATE TABLE `grades` (  `SID` varchar(25),  `Exam_ID` varchar(25) ,  `percentage`
decimal(5,2) CHECK (`percentage` between 0 and 100),  `grade` varchar(1) CHECK
(grade in ('A','B','C','D','E','F','a','b','c','d','e','f')), PRIMARY KEY(SID, Exam_ID) FOREIGN
KEY (`SID`) REFERENCES `student` (`SID`) ON UPDATE CASCADE ON DELETE
CASCADE,   FOREIGN KEY (`Exam_ID`) REFERENCES `exam` (`Exam_ID`) ON UPDATE
CASCADE ON DELETE CASCADE )
```

```sql
create table faculty(Sub_ID varchar(50), TID varchar(25), FOREIGN key(Sub_ID)
REFERENCES subject(Sub_ID) ON UPDATE CASCADE ON DELETE CASCADE, FOREIGN
key(TID) REFERENCES teacher(TID) ON UPDATE CASCADE ON DELETE CASCADE,
PRIMARY  key(Sub_ID,TID));
```

```sql
CREATE table class_has_exam(ClassID int(5),Exam_ID varchar(25), PRIMARY
key(ClassID, Exam_ID), FOREIGN key(ClassID) REFERENCES class(ClassID) ON UPDATE
CASCADE ON DELETE CASCADE, FOREIGN key(Exam_ID) REFERENCES exam(Exam_ID)
ON UPDATE CASCADE ON DELETE CASCADE);
```

```sql
create table teaches(TID varchar(25), ClassID int(5), PRIMARY key(TID, ClassID),
FOREIGN key(TID) REFERENCES teacher(TID) ON UPDATE CASCADE ON DELETE
CASCADE, FOREIGN key(ClassID) REFERENCES class(ClassID) ON UPDATE CASCADE ON
DELETE CASCADE);
```

Populating the Database

```
insert into class values(10,'A',2022,10001), (10,'B',2022,10002), (9,'A',2022,09001),
(10,'C',2022,10003), (9,'B',2022,9002), (9,'C',2022,9003), (8,'A',2022,8001),
(8,'B',2022,8002), (8,'C',2022,8003);
```

```
load data INFILE "student.csv" INTO TABLE student COLUMNS TERMINATED BY ','
OPTIONALLY ENCLOSED by " " ESCAPED by " " LINES TERMINATED by '\n';
```

```
insert into parent_lg values('22CS10A001', 'Jehan' , 'Mistry' , ‘jm6789@gmail.com' ,
'Father','Kanakapura Rd , JP Nagar 6th Phase','7423689','Engineer'), ('22CS10A002',
'Raj' , 'Khanna' , 'rajkhanna@gmail.com' , 'Father','Street
2','8652490192','Businessman'), ('22CS10A003', 'Sanjay' , 'Fernandez' ,
'sanjayspdf@gmail.com' , 'Father','Street 3','9937488392','Businessman'),
('22CS10B003', 'Manjot' , 'Singh' , 'singh23manjot@gmail.com' , 'Father','Street
4','9079423689','Engineer'), ('22CS09B003', 'Gurukiran' , 'Shetty' ,
'gk2098@gmail.com' , 'Father','Street 2','9232978901','Doctor'), ('22CS10A001',
'Smriti' , 'Mistry' , 'sm6789@gmail.com' , 'Mother','Kanakapura Rd , JP Nagar 6th
Phase','9103726491','Engineer'), ('22CS10A002', 'Suketha' , 'Khanna' ,
'skfdkhsk9@gmail.com' , 'Mother','Street 2','8652490192','Housewife'),
('22CS10A003', 'Nichole' , 'Fernandez' , 'nk@gmail.com' , 'Mother','Street
3','9218039480','Lawyer'), ('22CS10B003', 'Mannu' , 'Singh' , 'manny@gmail.com' ,
'Mother','Street 4','8794357290','Clerk')
```

```
INSERT into teacher VALUES ('TID001','Nagaraj','Bhupati','Street
1','560051','35792979', '2013-06-09'), ('TID002','Waseem','Jafferey','Street
3','560002','9907436932', '2015-07-11'), ('TID003','Anushka','Sheikh','Street
4','560051','8654392103', '2012-11-03'), ('TID004','Alakh','Pandya','Street
5','560102','9987654432', '2019-10-09'), ('TID005','Jeffrey','Lobo','Street
2','560001','9128902373', '2016-01-05'), ('TID006','Vinaya','Pai','Street
1','560051','9747320987', '2011-03-09');
```

```
insert into subject values('SUBID001','Chemistry','Core subject', ' the scientific study
of the properties and behavior of matter'), ('SUBID002','Physics','Core subject',
'Physics is the natural science that studies matter, its fundamental constituents, its
motion and behavior through space and time, and the related entities of energy and
force.'), ('SUBID003','Biology','Core subject', 'Study of living things and their vital
processes'),('SUBID004','Mathematics','Core subject', 'Topics as numbers, formulas
and related structures, shapes and the spaces in which they are
contained'),('SUBID005','History','Core subject', 'The study of the past. This study
would also include learning about humanities and social
sciences.'),('SUBID006','Geography','Core subject', 'The study of places and the
relationships between people and their environments'),('SUBID007','Civics','Core
subject', 'The study of the rights and obligations of citizens in
society.'),('SUBID008','Economics','Core subject', 'Study to determine the most logical
and effective use of resources to meet private and social
goals.'),('SUBID009','English','Language', 'Comprehension, grammar, vocabulary
pertaining to the English language'),('SUBID010','Hindi','Language', 'Comprehension,
grammar, vocabulary pertaining to the Hindi
language'),('SUBID011','Kannada','Language', 'Comprehension, grammar, vocabulary
pertaining to the Kannada language');
```

```
INSERT into exam values('EXAMID01','ISA-3','ISA ', '2022-04-04’, ’15:10:10',
'SUBID001'), ('EXAMID02','ISA-3','ISA','2022-04-05’, ’15:10:10','SUBID002'),
('EXAMID03','ISA-3','ISA','2022-04-06’, ’15:10:10','SUBID003'), ('EXAMID04','ISA-
3','ISA','2022-04-07’, ’15:10:10','SUBID004'), ('EXAMID05','ISA-3','ISA','2022-04-08’,
’13:10:10','SUBID005'), ('EXAMID06','ISA-3','ISA','2022-04-08’, ’15:10:10','SUBID006'),
('EXAMID07','ISA-3','ISA','2022-04-09’, ’13:10:10','SUBID007'), ('EXAMID08','ISA-
3','ISA','2022-04-09’, ’15:10:10','SUBID008'), ('EXAMID09','ISA-3','ISA','2022-04-10’,
’15:10:10','SUBID009'), ('EXAMID10','ISA-3','ISA','2022-04-11’, ’13:10:10','SUBID010'),
('EXAMID11','ISA-3','ISA','2022-04-11’, ’15:10:10','SUBID011');
```

```
INSERT into class_has_exam VALUES(10001,'EXAMID01'), (10002,'EXAMID01'),
(10003,'EXAMID01'), (10001,'EXAMID02'), (10002,'EXAMID02'), (10003,'EXAMID02'),
(10001,'EXAMID03'), (10002,'EXAMID03'), (10003,'EXAMID03'), (10001,'EXAMID04'),
(10002,'EXAMID04'), (10003,'EXAMID04'), (10001,'EXAMID05'), (10002,'EXAMID05'),
(10003,'EXAMID05'), (10001,'EXAMID06'), (10002,'EXAMID06'), (10003,'EXAMID06'),
(10001,'EXAMID07'), (10002,'EXAMID07'), (10003,'EXAMID07'), (10001,'EXAMID08'),
(10002,'EXAMID08'), (10003,'EXAMID08'), (10001,'EXAMID09'), (10002,'EXAMID09'),
(10003,'EXAMID09'), (10001,'EXAMID10'), (10002,'EXAMID10'), (10003,'EXAMID10'),
(10001,'EXAMID11'), (10002,'EXAMID11'), (10003,'EXAMID11');
```

```
insert into faculty values ('SUBID001','TID002'), ('SUBID003','TID002'),
('SUBID002','TID001'), ('SUBID001','TID001'), ('SUBID005','TID003'),
('SUBID006','TID003'), ('SUBID007','TID003'), ('SUBID008','TID003'),
('SUBID002','TID004'), ('SUBID004','TID004'), ('SUBID009','TID003'),
('SUBID010','TID004'), ('SUBID011','TID005'), ('SUBID010','TID006'),
('SUBID004','TID006');
```

```
load data INFILE "grade.csv" INTO TABLE grades COLUMNS TERMINATED BY ','
OPTIONALLY ENCLOSED by " " ESCAPED by " " LINES TERMINATED by '\n';
```

```
load data INFILE "achievements.csv" INTO TABLE achievements COLUMNS
TERMINATED BY ',' OPTIONALLY ENCLOSED by " " ESCAPED by " " LINES TERMINATED
by '\n';
```

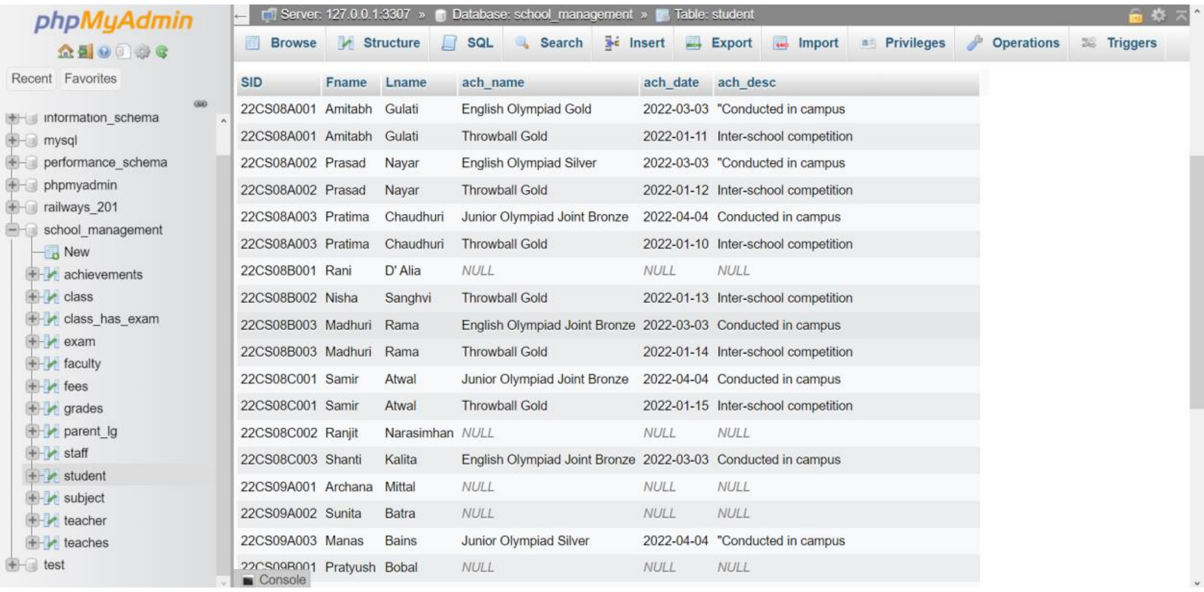
```
load data INFILE "teaches.csv" INTO TABLE teaches COLUMNS TERMINATED BY ','
OPTIONALLY ENCLOSED by " " ESCAPED by " " LINES TERMINATED by '\n';

load data INFILE "fees.csv" INTO TABLE fees COLUMNS TERMINATED BY ','
OPTIONALLY ENCLOSED by " " ESCAPED by " " LINES TERMINATED by '\n';
```

Join Queries

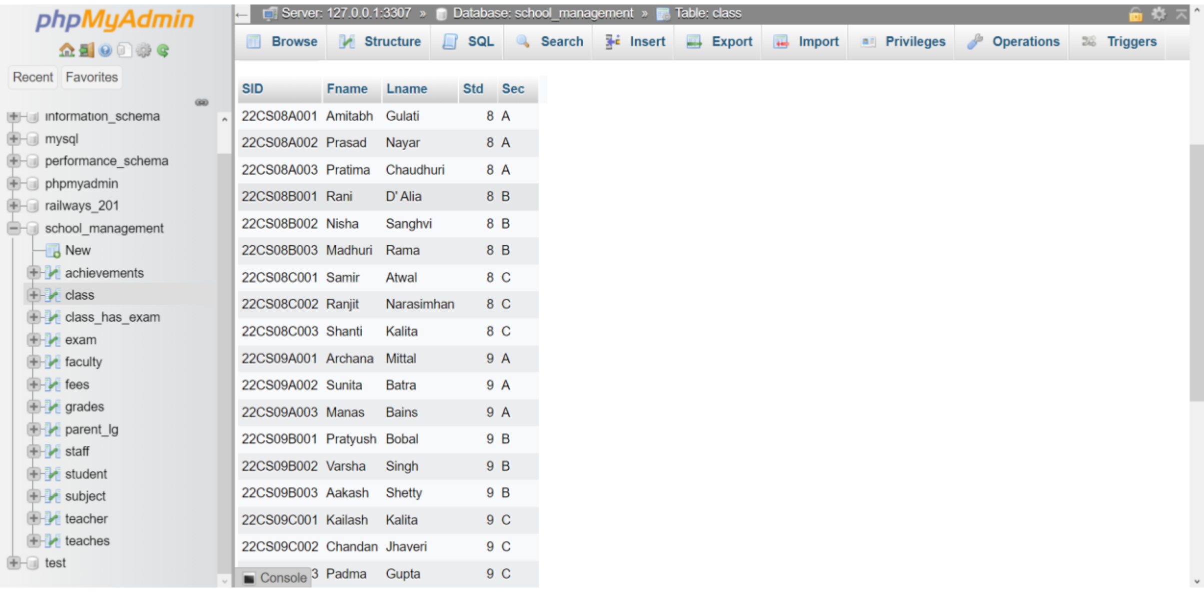
Showcase at least 4 join queries Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

1) Query to display the achievements of each and every student:
   select student.SID, Fname, Lname, ach_name, ach_date, ach_desc from student LEFT OUTER JOIN achievements on student.SID=achievements.SID;



SID	Fname	Lname	ach_name	ach_date	ach_desc
22CS08A001	Amitabh	Gulati	English Olympiad Gold	2022-03-03	"Conducted in campus
22CS08A001	Amitabh	Gulati	Throwball Gold	2022-01-11	Inter-school competition
22CS08A002	Prasad	Nayar	English Olympiad Silver	2022-03-03	"Conducted in campus
22CS08A002	Prasad	Nayar	Throwball Gold	2022-01-12	Inter-school competition
22CS08A003	Pratima	Chaudhuri	Junior Olympiad Joint Bronze	2022-04-04	Conducted in campus
22CS08A003	Pratima	Chaudhuri	Throwball Gold	2022-01-10	Inter-school competition
22CS08B001	Rani	D' Alia	NULL	NULL	NULL
22CS08B002	Nisha	Sanghvi	Throwball Gold	2022-01-13	Inter-school competition
22CS08B003	Madhuri	Rama	English Olympiad Joint Bronze	2022-03-03	Conducted in campus
22CS08B003	Madhuri	Rama	Throwball Gold	2022-01-14	Inter-school competition
22CS08C001	Samir	Atwal	Junior Olympiad Joint Bronze	2022-04-04	Conducted in campus
22CS08C001	Samir	Atwal	Throwball Gold	2022-01-15	Inter-school competition
22CS08C002	Ranjit	Narasimhan	NULL	NULL	NULL
22CS08C003	Shanti	Kalita	English Olympiad Joint Bronze	2022-03-03	Conducted in campus
22CS09A001	Archana	Mittal	NULL	NULL	NULL
22CS09A002	Sunita	Batra	NULL	NULL	NULL
22CS09A003	Manas	Bains	Junior Olympiad Silver	2022-04-04	"Conducted in campus
22CS09B001	Pratyush	Bobal	NULL	NULL	NULL

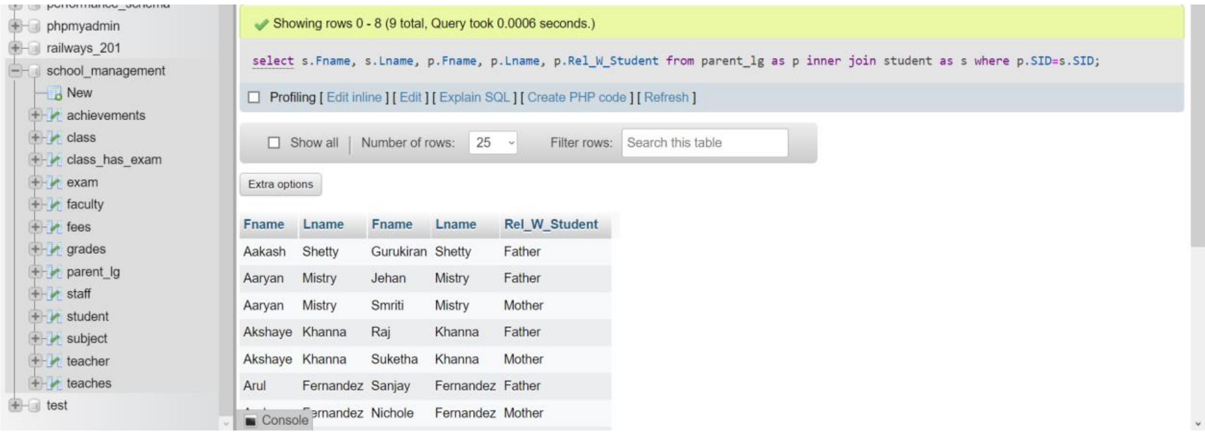
2) Query to display the class which each student is studying in:
   select SID, Fname, Lname, Std, Sec from class natural join student;



SID	Fname	Lname	Std	Sec
22CS08A001	Amitabh	Gulati	8	A
22CS08A002	Prasad	Nayar	8	A
22CS08A003	Pratima	Chaudhuri	8	A
22CS08B001	Rani	D' Alia	8	B
22CS08B002	Nisha	Sanghvi	8	B
22CS08B003	Madhuri	Rama	8	B
22CS08C001	Samir	Atwal	8	C
22CS08C002	Ranjit	Narasimhan	8	C
22CS08C003	Shanti	Kalita	8	C
22CS09A001	Archana	Mittal	9	A
22CS09A002	Sunita	Batra	9	A
22CS09A003	Manas	Bains	9	A
22CS09B001	Pratyush	Bobal	9	B
22CS09B002	Varsha	Singh	9	B
22CS09B003	Aakash	Shetty	9	B
22CS09C001	Kailash	Kalita	9	C
22CS09C002	Chandan	Jhaveri	9	C
3	Padma	Gupta	9	C

3) Query to list the parents of all the students:

select s.Fname, s.Lname, p.Fname, p.Lname, p.Rel_W_Student from parent_lg as p inner join student as s where p.SID=s.SID;



✓ Showing rows 0 - 8 (9 total, Query took 0.0006 seconds.)

```
select s.Fname, s.Lname, p.Fname, p.Lname, p.Rel_W_Student from parent_lg as p inner join student as s where p.SID=s.SID;
```

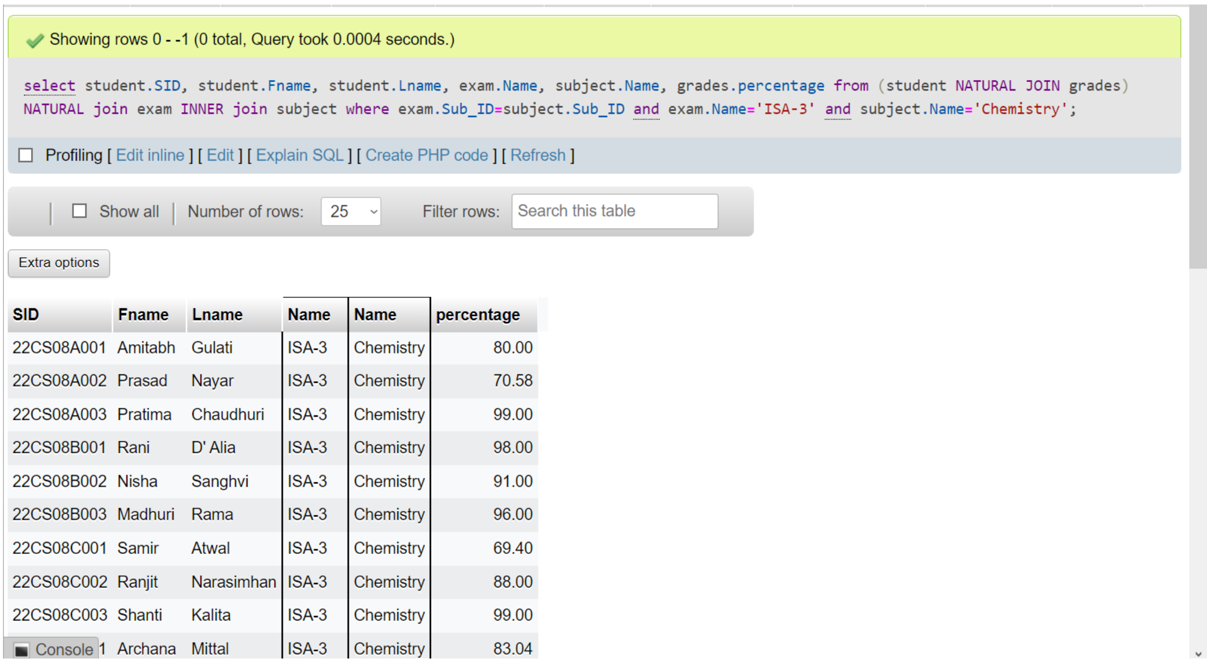
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table

Extra options

Fname	Lname	Fname	Lname	Rel_W_Student
Aakash	Shetty	Gurukiran	Shetty	Father
Aaryan	Mistry	Jehan	Mistry	Father
Aaryan	Mistry	Smriti	Mistry	Mother
Akshaye	Khanna	Raj	Khanna	Father
Akshaye	Khanna	Suketha	Khanna	Mother
Arul	Fernandez	Sanjay	Fernandez	Father
	ernandez	Nichole	Fernandez	Mother

4) Query to display every student and their Chemistry marks in ISA-3:
select student.SID, student.Fname, student.Lname, exam.Name, subject.Name, grades.percentage from (student NATURAL JOIN grades) NATURAL join exam INNER join subject where exam.Sub_ID=subject.Sub_ID and exam.Name='ISA-3' and subject.Name='Chemistry';



✓ Showing rows 0 - -1 (0 total, Query took 0.0004 seconds.)

```
select student.SID, student.Fname, student.Lname, exam.Name, subject.Name, grades.percentage from (student NATURAL JOIN grades)
NATURAL join exam INNER join subject where exam.Sub_ID=subject.Sub_ID and exam.Name='ISA-3' and subject.Name='Chemistry';
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table

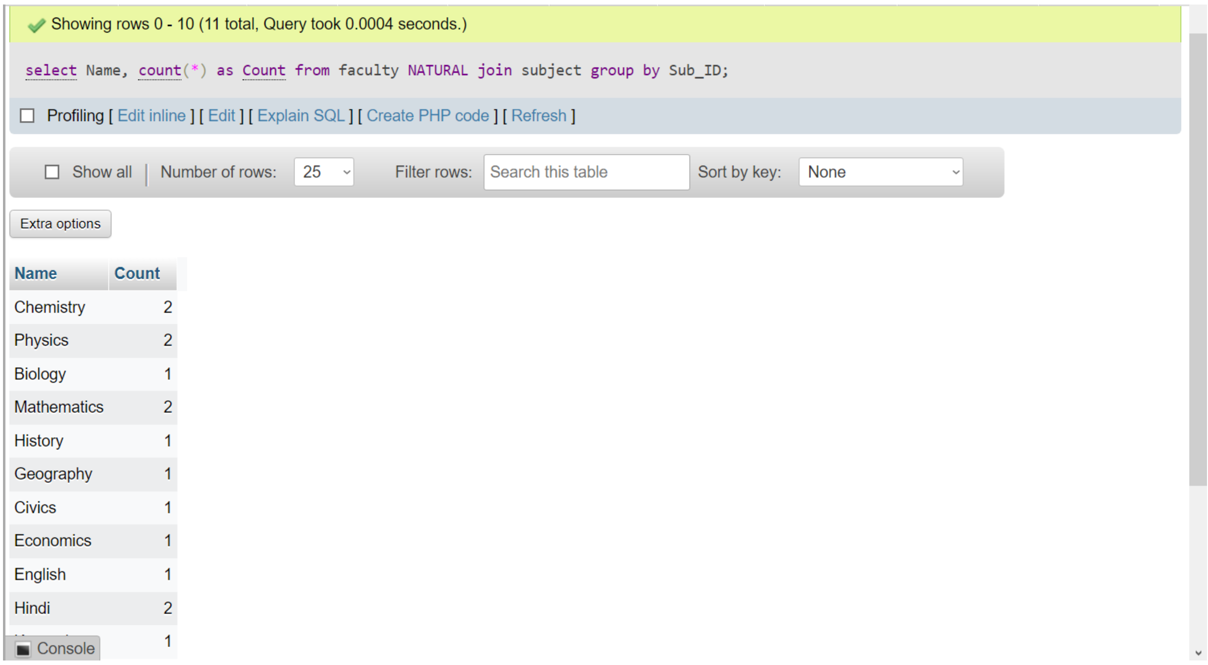
Extra options

SID	Fname	Lname	Name	Name	percentage
22CS08A001	Amitabh	Gulati	ISA-3	Chemistry	80.00
22CS08A002	Prasad	Nayar	ISA-3	Chemistry	70.58
22CS08A003	Pratima	Chaudhuri	ISA-3	Chemistry	99.00
22CS08B001	Rani	D' Alia	ISA-3	Chemistry	98.00
22CS08B002	Nisha	Sanghvi	ISA-3	Chemistry	91.00
22CS08B003	Madhuri	Rama	ISA-3	Chemistry	96.00
22CS08C001	Samir	Atwal	ISA-3	Chemistry	69.40
22CS08C002	Ranjit	Narasimhan	ISA-3	Chemistry	88.00
22CS08C003	Shanti	Kalita	ISA-3	Chemistry	99.00
1	Archana	Mittal	ISA-3	Chemistry	83.04

Aggregate Functions

Showcase at least 4 Aggregate function queries Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

1) Query to display the number of teachers for every subject:
   select Name, count(*) as Count from faculty NATURAL join subject group by Sub_ID;



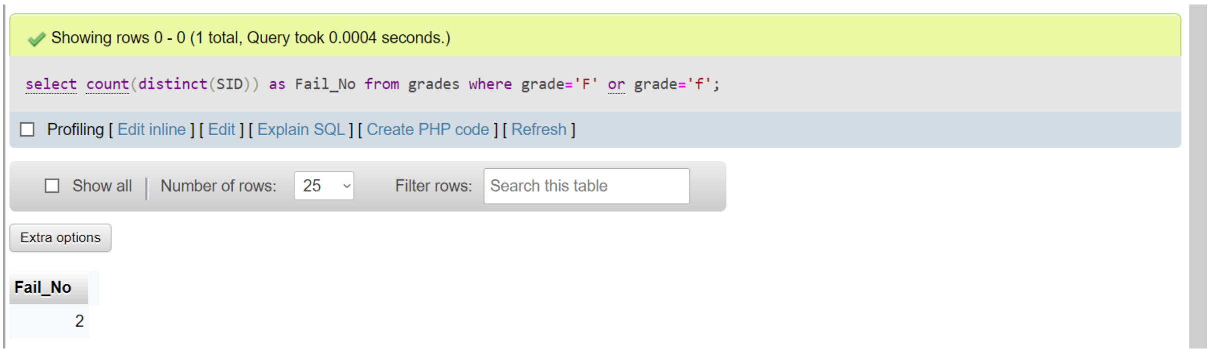
✓ Showing rows 0 - 10 (11 total, Query took 0.0004 seconds.)

```
select Name, count(*) as Count from faculty NATURAL join subject group by Sub_ID;
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
☐ Show all
Number of rows: 25
Filter rows: Search this table
Sort by key: None
Extra options
Console

Name	Count
Chemistry	2
Physics	2
Biology	1
Mathematics	2
History	1
Geography	1
Civics	1
Economics	1
English	1
Hindi	2
	1

2) Query to display the number of students who have failed a test:
   select count(distinct(SID)) as Fail_No from grades where percentage<=35;



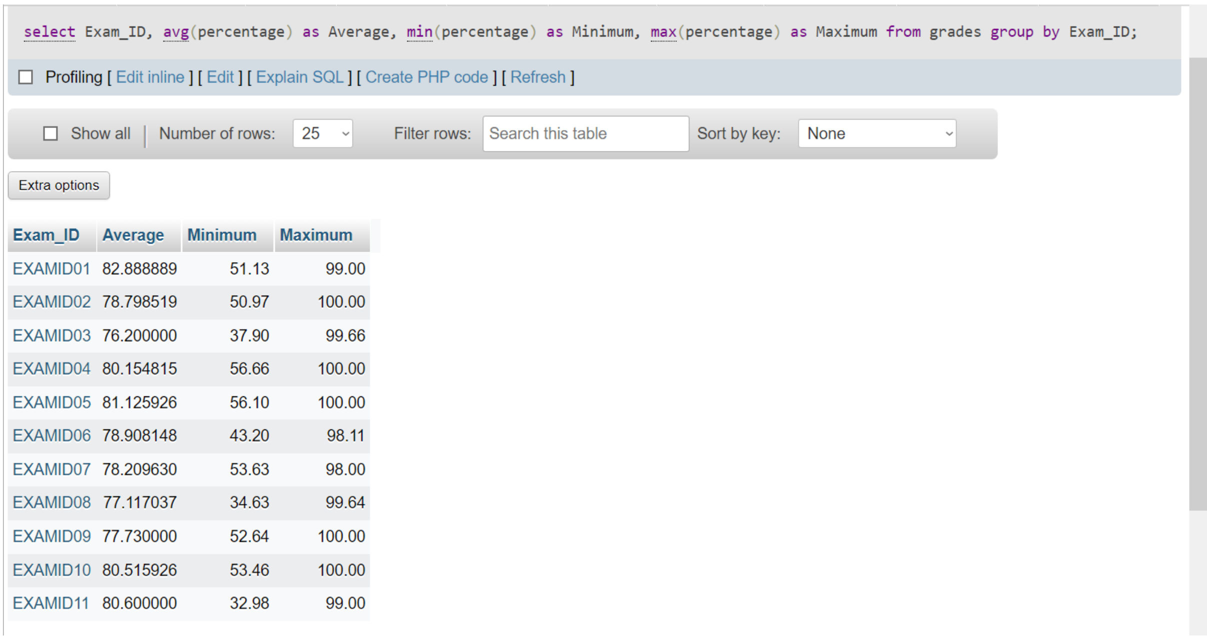
✓ Showing rows 0 - 0 (1 total, Query took 0.0004 seconds.)

```
select count(distinct(SID)) as Fail_No from grades where grade='F' or grade='f';
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
☐ Show all
Number of rows: 25
Filter rows: Search this table
Extra options

Fail_No
2

3) Query to display the average, minimum, and maximum scores (in %) of each test:
   select Exam_ID, avg(percentage) as Average, min(percentage) as Minimum, max(percentage) as Maximum from grades group by Exam_ID;

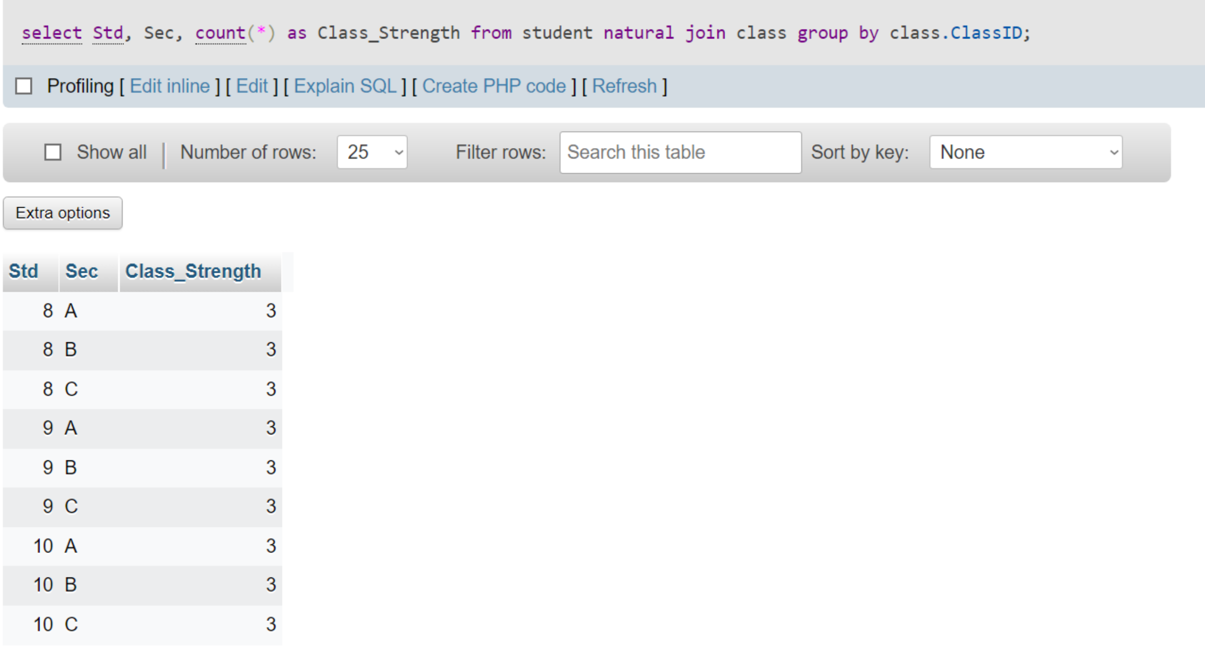


```
select Exam_ID, avg(percentage) as Average, min(percentage) as Minimum, max(percentage) as Maximum from grades group by Exam_ID;
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
☐ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options

Exam_ID	Average	Minimum	Maximum
EXAMID01	82.888889	51.13	99.00
EXAMID02	78.798519	50.97	100.00
EXAMID03	76.200000	37.90	99.66
EXAMID04	80.154815	56.66	100.00
EXAMID05	81.125926	56.10	100.00
EXAMID06	78.908148	43.20	98.11
EXAMID07	78.209630	53.63	98.00
EXAMID08	77.117037	34.63	99.64
EXAMID09	77.730000	52.64	100.00
EXAMID10	80.515926	53.46	100.00
EXAMID11	80.600000	32.98	99.00

4) Query to display the strength of each class:
select Std, Sec, count(*) as Class_Strength from student natural join class group by class.ClassID;



```
select Std, Sec, count(*) as Class_Strength from student natural join class group by class.ClassID;
```

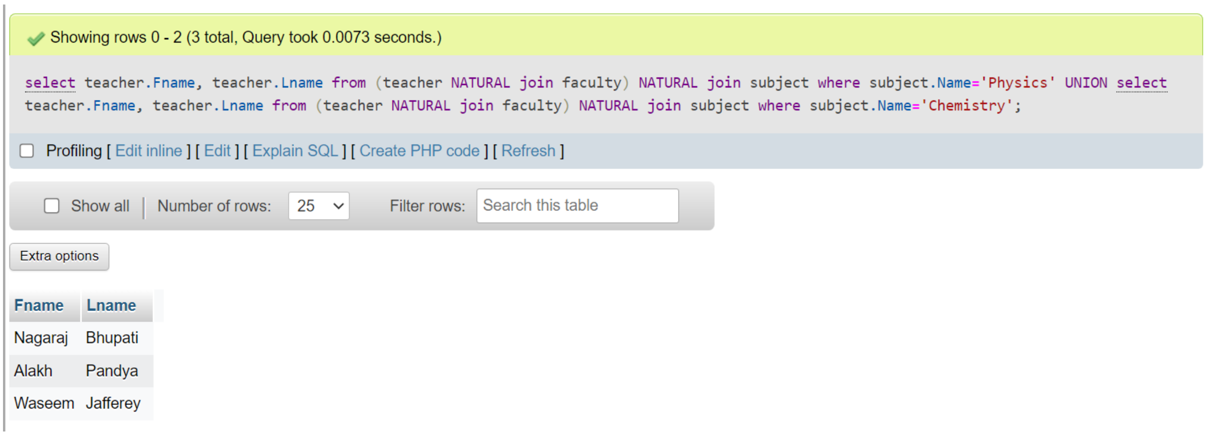
☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
☐ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options

Std	Sec	Class_Strength
8	A	3
8	B	3
8	C	3
9	A	3
9	B	3
9	C	3
10	A	3
10	B	3
10	C	3

Set Operations

Showcase at least 4 Set Operations queries Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

1) Query to find the list of teachers who teach either chemistry or physics:
   select teacher.Fname, teacher.Lname from (teacher NATURAL join faculty) NATURAL join subject where subject.Name='Physics'
   UNION
   select teacher.Fname, teacher.Lname from (teacher NATURAL join faculty) NATURAL join subject where subject.Name='Chemistry';



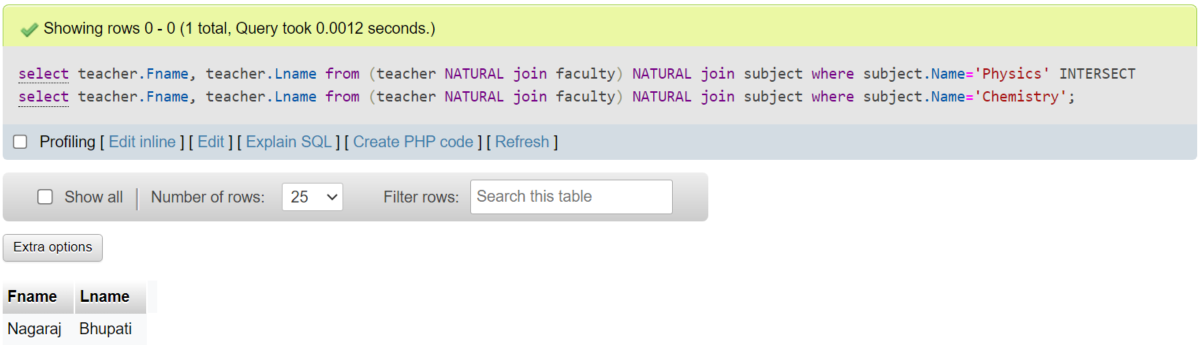
✓ Showing rows 0 - 2 (3 total, Query took 0.0073 seconds.)

```
select teacher.Fname, teacher.Lname from (teacher NATURAL join faculty) NATURAL join subject where subject.Name='Physics' UNION select
teacher.Fname, teacher.Lname from (teacher NATURAL join faculty) NATURAL join subject where subject.Name='Chemistry';
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table
Extra options

Fname	Lname
Nagaraj	Bhupati
Alakh	Pandya
Waseem	Jafferey

2) Query to find the list of teachers who teach both chemistry and physics:
   select teacher.Fname, teacher.Lname from (teacher NATURAL join faculty) NATURAL join subject where subject.Name='Physics'
   INTERSECT
   select teacher.Fname, teacher.Lname from (teacher NATURAL join faculty) NATURAL join subject where subject.Name='Chemistry';



✓ Showing rows 0 - 0 (1 total, Query took 0.0012 seconds.)

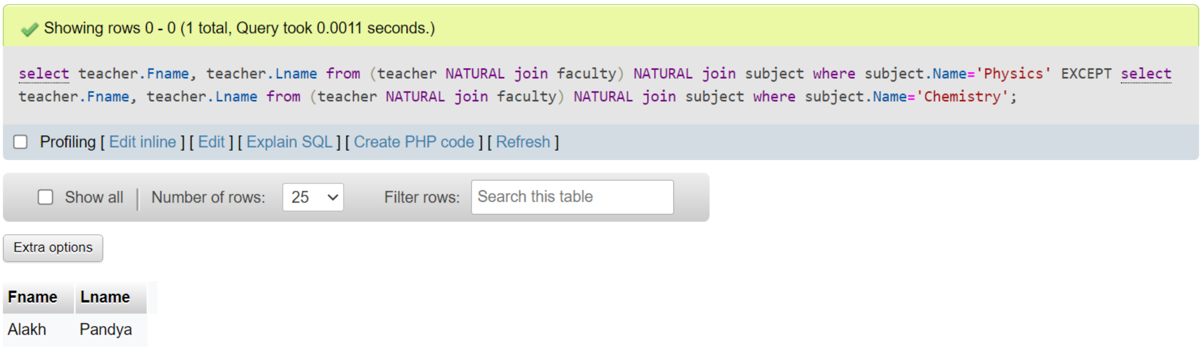
```
select teacher.Fname, teacher.Lname from (teacher NATURAL join faculty) NATURAL join subject where subject.Name='Physics' INTERSECT
select teacher.Fname, teacher.Lname from (teacher NATURAL join faculty) NATURAL join subject where subject.Name='Chemistry';
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table
Extra options

Fname	Lname
Nagaraj	Bhupati

3) Query to find the list of teachers who teach physics but not chemistry:
   select teacher.Fname, teacher.Lname from (teacher NATURAL join faculty) NATURAL join subject where subject.Name='Physics'

EXCEPT
select teacher.Fname, teacher.Lname from (teacher NATURAL join faculty) NATURAL join subject where subject.Name='Chemistry';



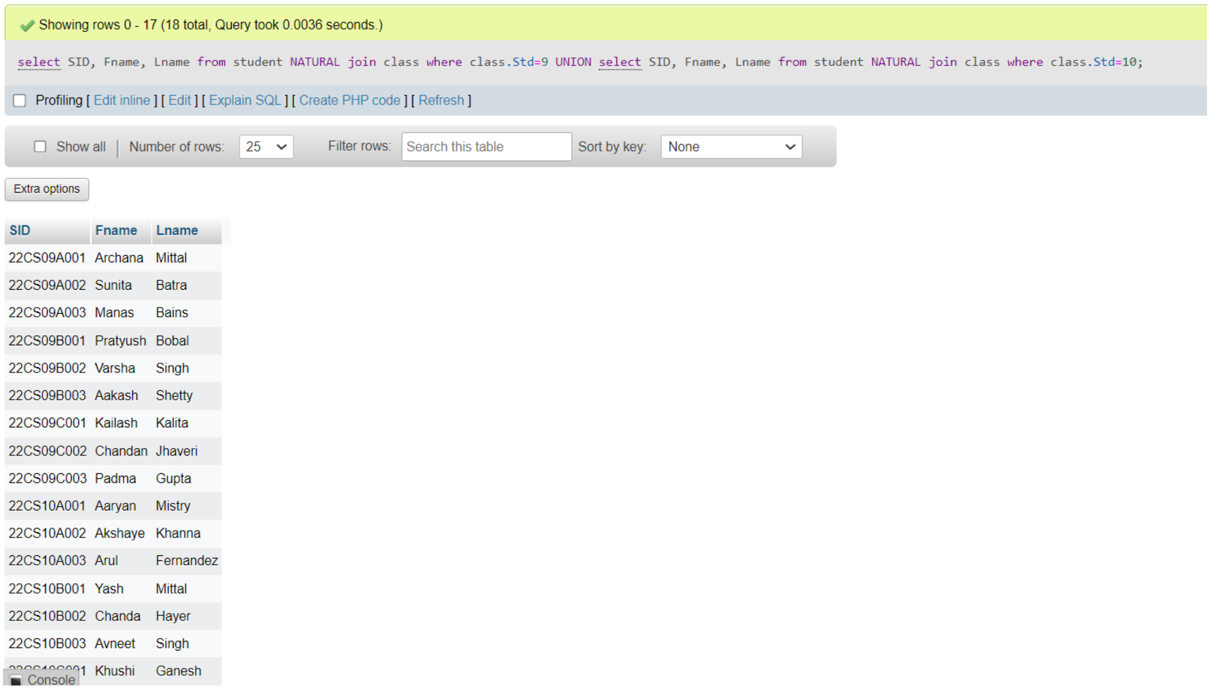
✓ Showing rows 0 - 0 (1 total, Query took 0.0011 seconds.)

```
select teacher.Fname, teacher.Lname from (teacher NATURAL join faculty) NATURAL join subject where subject.Name='Physics' EXCEPT select
teacher.Fname, teacher.Lname from (teacher NATURAL join faculty) NATURAL join subject where subject.Name='Chemistry';
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table
Extra options

Fname	Lname
Alakh	Pandya

4) Query to find the list of students who are appearing for their board exams (i.e. the list of students who are in class 9 or 10):
select SID, Fname, Lname from student NATURAL join class where class.Std=9
UNION
select SID, Fname, Lname from student NATURAL join class where class.Std=10;



✓ Showing rows 0 - 17 (18 total, Query took 0.0036 seconds.)

```
select SID, Fname, Lname from student NATURAL join class where class.Std=9 UNION select SID, Fname, Lname from student NATURAL join class where class.Std=10;
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options
Console

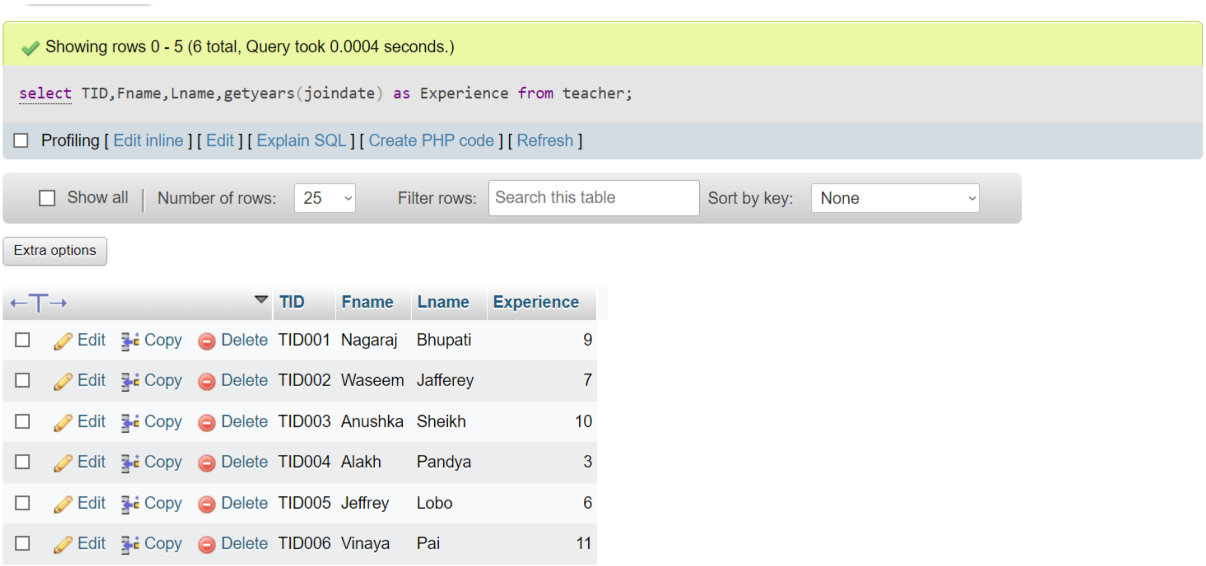
SID	Fname	Lname
22CS09A001	Archana	Mittal
22CS09A002	Sunita	Batra
22CS09A003	Manas	Bains
22CS09B001	Pratyush	Bobal
22CS09B002	Varsha	Singh
22CS09B003	Aakash	Shetty
22CS09C001	Kailash	Kalita
22CS09C002	Chandan	Jhaveri
22CS09C003	Padma	Gupta
22CS10A001	Aaryan	Mistry
22CS10A002	Akshaye	Khanna
22CS10A003	Arul	Fernandez
22CS10B001	Yash	Mittal
22CS10B002	Chanda	Hayer
22CS10B003	Avneet	Singh
22CS10C001	Khushi	Ganesh

Functions and Procedures

Create a Function and Procedure. State the objective of the function / Procedure. Run and display the results

1) Function to find experience (in years) of a teacher:

```
DELIMITER $$
CREATE FUNCTION getyears(data date) RETURNS int DETERMINISTIC
BEGIN
DECLARE currentDate DATE;
Select current_date()into currentDate;
RETURN (YEAR(currentDate)-YEAR(data));
END
$$
DELIMITER ;
select TID,Fname,Lname,getyears(joindate) as Experience from teacher;
```



✓ Showing rows 0 - 5 (6 total, Query took 0.0004 seconds.)

```
select TID,Fname,Lname,getyears(joindate) as Experience from teacher;
```

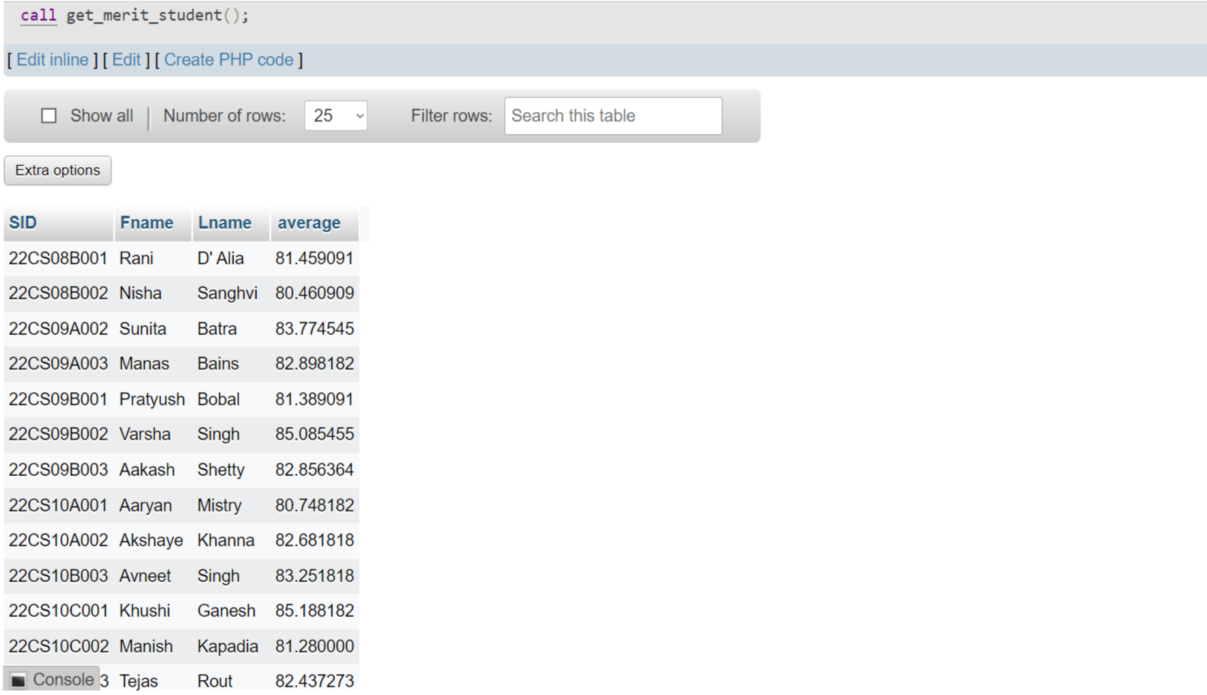
☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
☐ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options

←T→				TID	Fname	Lname	Experience
☐	Edit	Copy	Delete	TID001	Nagaraj	Bhupati	9
☐	Edit	Copy	Delete	TID002	Waseem	Jafferey	7
☐	Edit	Copy	Delete	TID003	Anushka	Sheikh	10
☐	Edit	Copy	Delete	TID004	Alakh	Pandya	3
☐	Edit	Copy	Delete	TID005	Jeffrey	Lobo	6
☐	Edit	Copy	Delete	TID006	Vinaya	Pai	11

2) Procedure to return the list of students who have achieved distinction, the criteria being the average percentage of the student over all the exams must be greater than 80:

```
create view avg_percentage as select SID, avg(percentage) as average from grades GROUP by SID;
DELIMITER &&
CREATE PROCEDURE get_merit_student ()
BEGIN
SELECT SID, Fname, Lname, average FROM avg_percentage NATURAL JOIN student WHERE average > 80;
```

```
END &&
DELIMITER ;
CALL get_merit_student();
```



```
call get_merit_student();
```

[Edit inline] [Edit] [Create PHP code]

□ Show all | Number of rows: 25
Filter rows: Search this table
Extra options

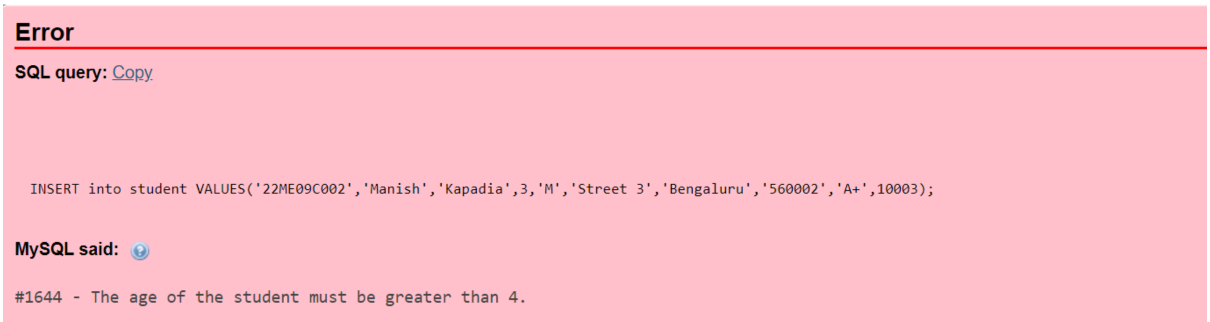
SID	Fname	Lname	average
22CS08B001	Rani	D' Alia	81.459091
22CS08B002	Nisha	Sanghvi	80.460909
22CS09A002	Sunita	Batra	83.774545
22CS09A003	Manas	Bains	82.898182
22CS09B001	Pratyush	Bobal	81.389091
22CS09B002	Varsha	Singh	85.085455
22CS09B003	Aakash	Shetty	82.856364
22CS10A001	Aaryan	Mistry	80.748182
22CS10A002	Akshaye	Khanna	82.681818
22CS10B003	Avneet	Singh	83.251818
22CS10C001	Khushi	Ganesh	85.188182
22CS10C002	Manish	Kapadia	81.280000
Console 3	Tejas	Rout	82.437273

Triggers and Cursors

Create a Trigger and a Cursor. State the objective. Run and display the results.

1) Trigger on insertion to the ‘student’ table when data for a new student gets added, to make sure that the age of the student is greater than 4:

```
DELIMITER $$
CREATE TRIGGER check_age
AFTER INSERT
ON student FOR EACH ROW
BEGIN
 DECLARE error_msg VARCHAR(255);
 declare val int;
 SET error_msg = ('The age of the student must be greater than 4.');
 SET val=(select Age from student where SID=new.SID);
 IF val<=4 THEN
 SIGNAL SQLSTATE '45000'
 SET MESSAGE_TEXT = error_msg;
 DELETE FROM student WHERE SID=new.SID;
 END IF;
END $$
DELIMITER ;
INSERT into student VALUES('22ME09C002','Manish','Kapadia',3,'M','Street 3','Bengaluru','560002','A+',10003);
```



Error

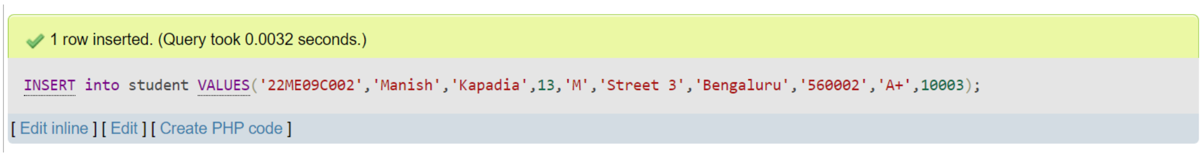
SQL query:

```
INSERT into student VALUES('22ME09C002','Manish','Kapadia',3,'M','Street 3','Bengaluru','560002','A+',10003);
```

MySQL said:

```
#1644 - The age of the student must be greater than 4.
```

```
INSERT into student VALUES('22ME09C002','Manish','Kapadia',13,'M','Street 3','Bengaluru','560002','A+',10003);
```



✓ 1 row inserted. (Query took 0.0032 seconds.)

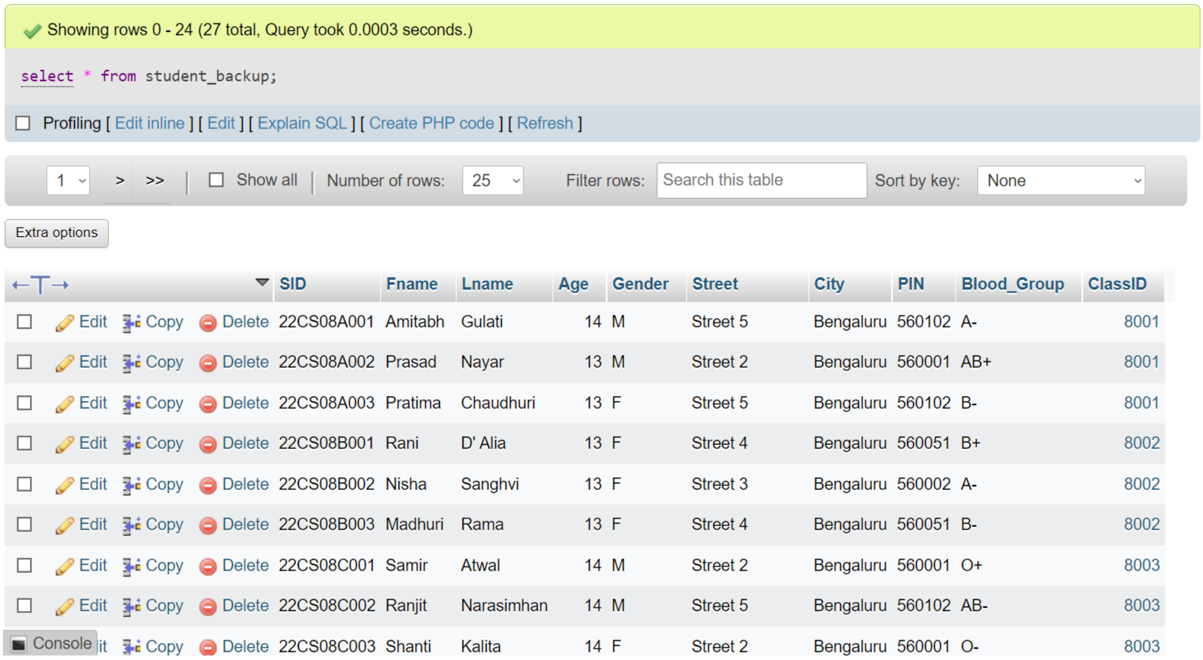
```
INSERT into student VALUES('22ME09C002','Manish','Kapadia',13,'M','Street 3','Bengaluru','560002','A+',10003);
```

2) Procedure that backups the contents of the student table to the backup table using cursors:

```
CREATE TABLE student_backup (SID varchar(25) NOT NULL, Fname
varchar(255), Lname varchar(255),  Age int(3), Gender varchar(225) CHECK
(Gender in
('M','F','m','f','Male','Female','male','female','MALE','FEMALE','Unknown','unk
nown','UNKNOWN')), Street varchar(255), City varchar(255), PIN varchar(15),
Blood_Group varchar(5), ClassID int(5), PRIMARY KEY(SID), FOREIGN
KEY(ClassID) REFERENCES class(ClassID));

DELIMITER //
CREATE PROCEDURE StudBackup()
BEGIN
DECLARE done INT DEFAULT 0;
DECLARE SID varchar(25);
DECLARE Fname, Lname, Gender, Street, City VARCHAR(255);
DECLARE Age int(3);
DECLARE ClassID int(5);
DECLARE PIN varchar(15);
DECLARE Blood_Group varchar(5);
DECLARE cur CURSOR FOR SELECT * FROM student;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
OPEN cur;
label: LOOP
FETCH cur INTO SID, Fname, Lname, Age, Gender, Street, City, PIN,
Blood_Group, ClassID;
INSERT INTO student_backup VALUES(SID, Fname, Lname, Age, Gender,
Street, City, PIN, Blood_Group, ClassID);
IF done = 1 THEN LEAVE label;
END IF;
END LOOP;
CLOSE cur;
END//
DELIMITER ;

call StudBackup();
select * from student_backup;
```



✓ Showing rows 0 - 24 (27 total, Query took 0.0003 seconds.)

```
select * from student_backup;
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

←T→				SID	Fname	Lname	Age	Gender	Street	City	PIN	Blood_Group	ClassID
□	Edit	Copy	Delete	22CS08A001	Amitabh	Gulati	14	M	Street 5	Bengaluru	560102	A-	8001
□	Edit	Copy	Delete	22CS08A002	Prasad	Nayar	13	M	Street 2	Bengaluru	560001	AB+	8001
□	Edit	Copy	Delete	22CS08A003	Pratima	Chaudhuri	13	F	Street 5	Bengaluru	560102	B-	8001
□	Edit	Copy	Delete	22CS08B001	Rani	D' Alia	13	F	Street 4	Bengaluru	560051	B+	8002
□	Edit	Copy	Delete	22CS08B002	Nisha	Sanghvi	13	F	Street 3	Bengaluru	560002	A-	8002
□	Edit	Copy	Delete	22CS08B003	Madhuri	Rama	13	F	Street 4	Bengaluru	560051	B-	8002
□	Edit	Copy	Delete	22CS08C001	Samir	Atwal	14	M	Street 2	Bengaluru	560001	O+	8003
□	Edit	Copy	Delete	22CS08C002	Ranjit	Narasimhan	14	M	Street 5	Bengaluru	560102	AB-	8003
Console	it	Copy	Delete	22CS08C003	Shanti	Kalita	14	F	Street 2	Bengaluru	560001	O-	8003

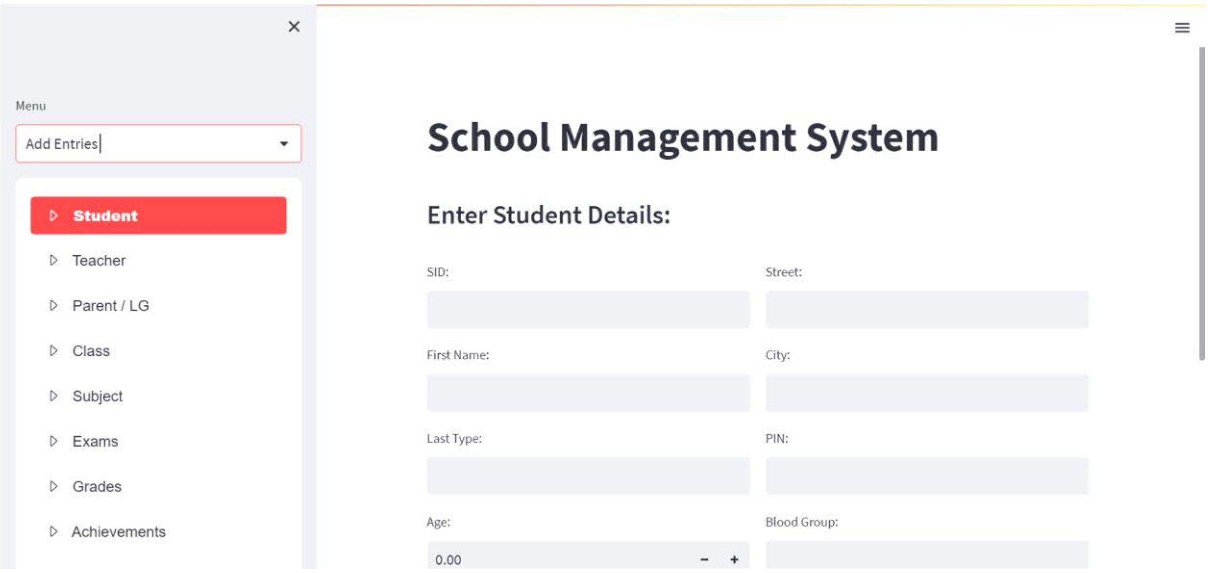
Developing a Frontend

The frontend should support

1. Addition, Modification and Deletion of records from any chosen table
2. There should be a window to accept and run any SQL statement and display the result

The frontend is designed using Streamlit. It supports insertion and deletion of records for all 11 tables, and modification of records for the student, teacher, and grades table. It can also display all the records in every table. Moreover, there is a window in which the user can run an SQL query and view the result.

Add Entries:



School Management System

Enter Student Details:

SID:

Street:

First Name:

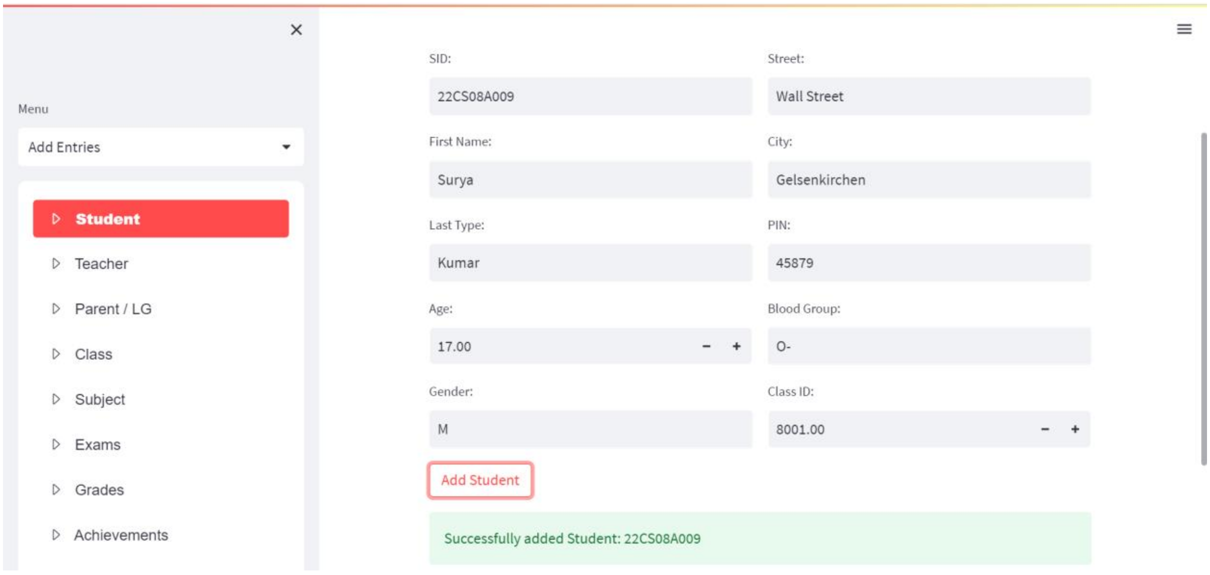
City:

Last Type:

PIN:

Age: 0.00

Blood Group:



SID: 22CS08A009

Street: Wall Street

First Name: Surya

City: Gelsenkirchen

Last Type: Kumar

PIN: 45879

Age: 17.00

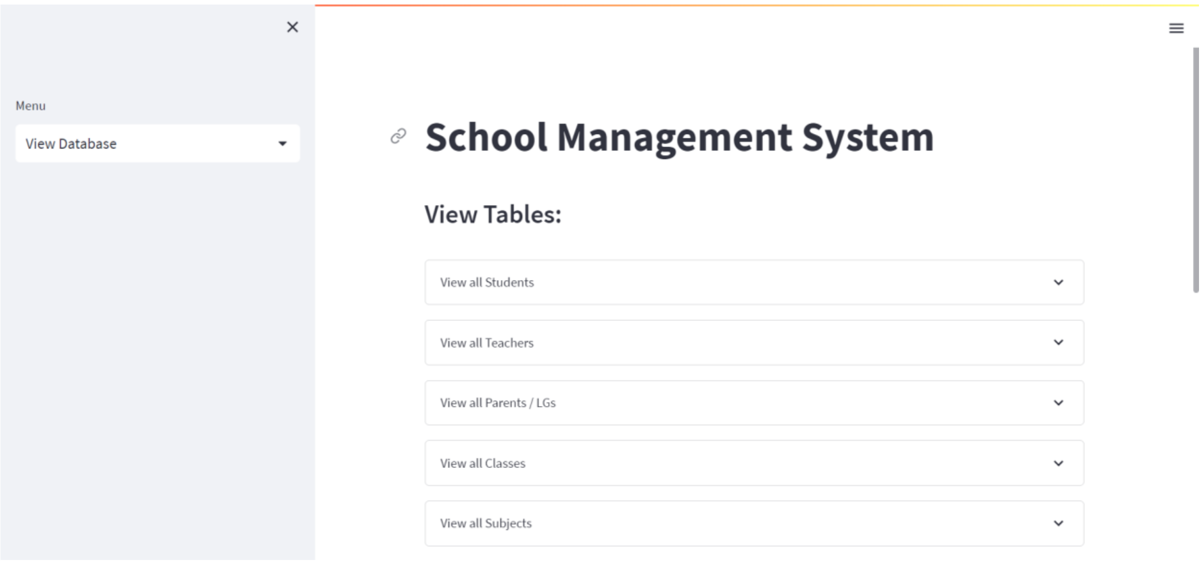
Blood Group: O-

Gender: M

Class ID: 8001.00

Successfully added Student: 22CS08A009

View Tables:

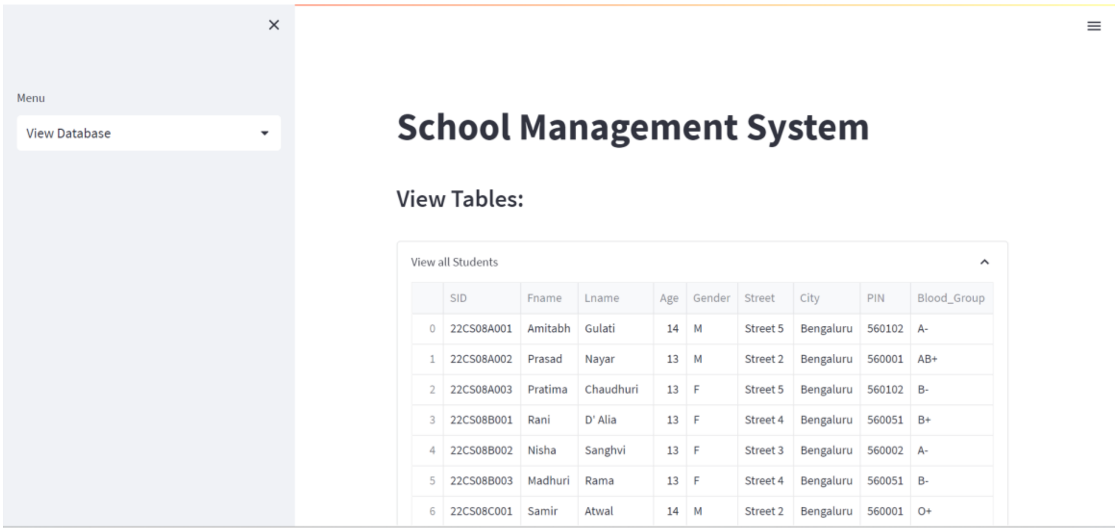


Menu

View Database

School Management System

View Tables:



Menu

View Database

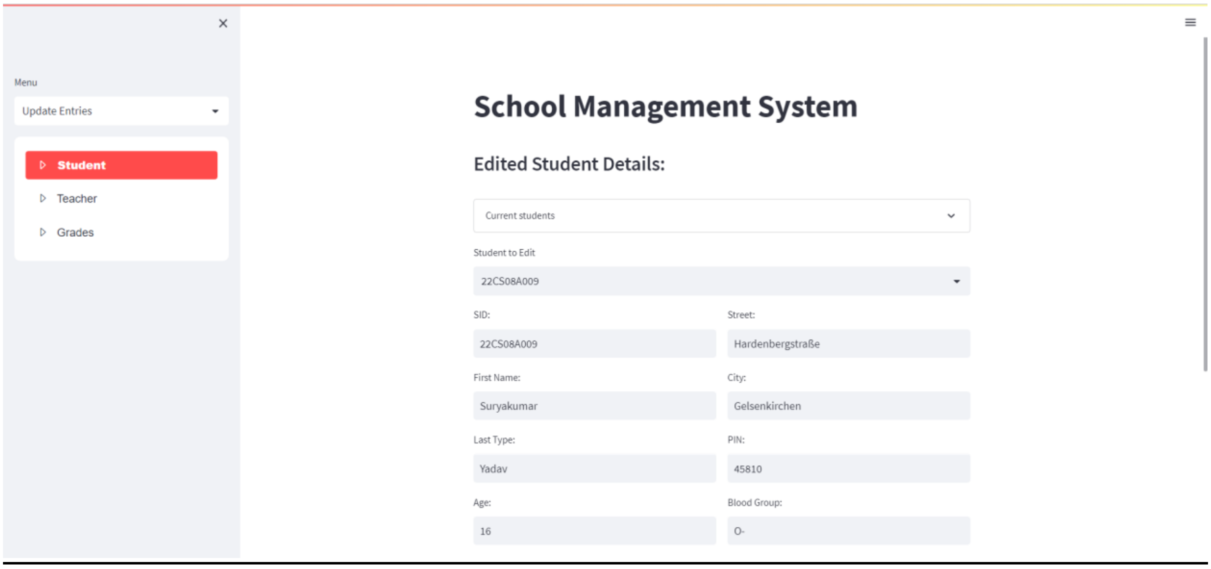
School Management System

View Tables:

View all Students

	SID	Fname	Lname	Age	Gender	Street	City	PIN	Blood_Group
0	22CS08A001	Amitabh	Gulati	14	M	Street 5	Bengaluru	560102	A-
1	22CS08A002	Prasad	Nayar	13	M	Street 2	Bengaluru	560001	AB+
2	22CS08A003	Pratima	Chaudhuri	13	F	Street 5	Bengaluru	560102	B-
3	22CS08B001	Rani	D' Alia	13	F	Street 4	Bengaluru	560051	B+
4	22CS08B002	Nisha	Sanghvi	13	F	Street 3	Bengaluru	560002	A-
5	22CS08B003	Madhuri	Rama	13	F	Street 4	Bengaluru	560051	B-
6	22CS08C001	Samir	Atwal	14	M	Street 2	Bengaluru	560001	O+

Update Entries:



Menu

Update Entries

School Management System

Edited Student Details:

Current students

Student to Edit

22CS08A009

SID: 22CS08A009

Street: Hardenbergstraße

First Name: Suryakumar

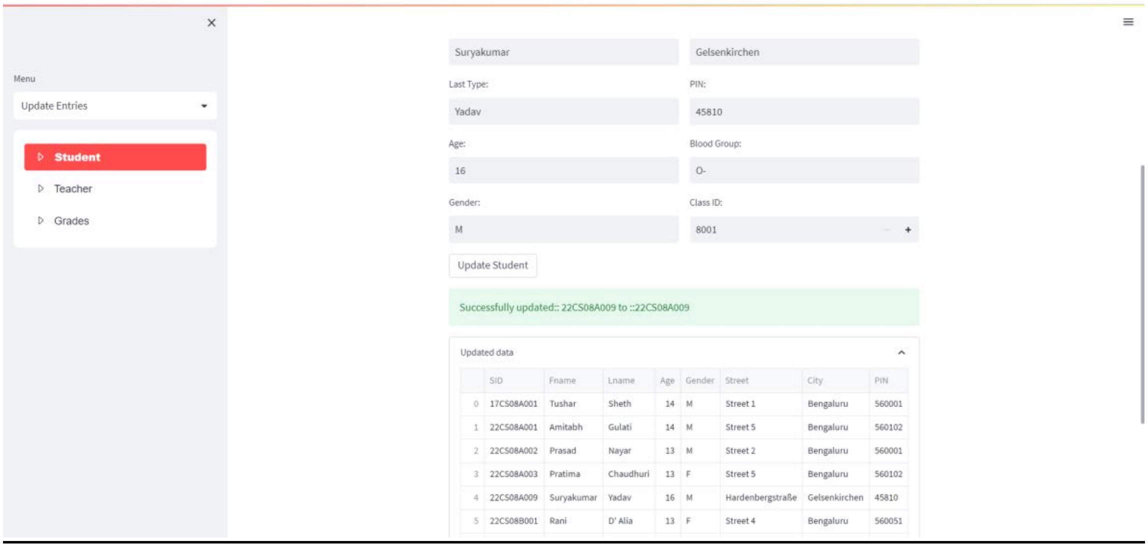
City: Gelsenkirchen

Last Type: Yadav

PIN: 45810

Age: 16

Blood Group: O-



Suryakumar

Gelsenkirchen

Last Type: Yadav

PIN: 45810

Age: 16

Blood Group: O-

Gender: M

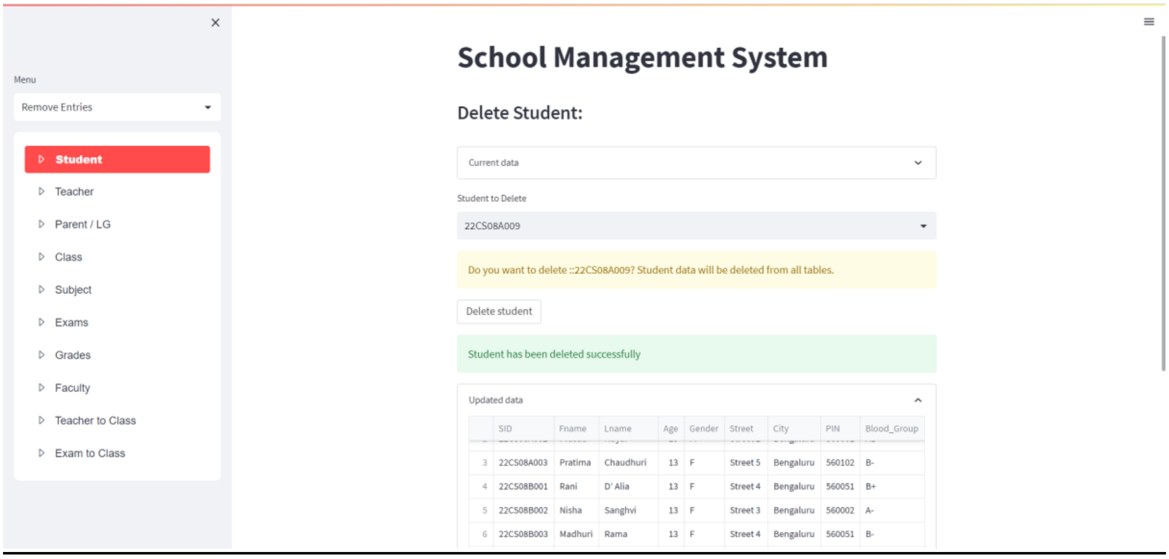
Class ID: 8001

Successfully updated:: 22CS08A009 to ::22CS08A009

Updated data

	SID	Fname	Lname	Age	Gender	Street	City	PIN
0	17CS08A001	Tushar	Sheth	14	M	Street 1	Bengaluru	560001
1	22CS08A001	Amitabh	Gulati	14	M	Street 5	Bengaluru	560102
2	22CS08A002	Prasad	Nayar	13	M	Street 2	Bengaluru	560001
3	22CS08A003	Pratima	Chaudhuri	13	F	Street 5	Bengaluru	560102
4	22CS08A009	Suryakumar	Yadav	16	M	Hardenbergstraße	Gelsenkirchen	45810
5	22CS08B001	Rani	D' Alia	13	F	Street 4	Bengaluru	560051

Remove Entries:



School Management System

Delete Student:

Student to Delete: 22CS08A009

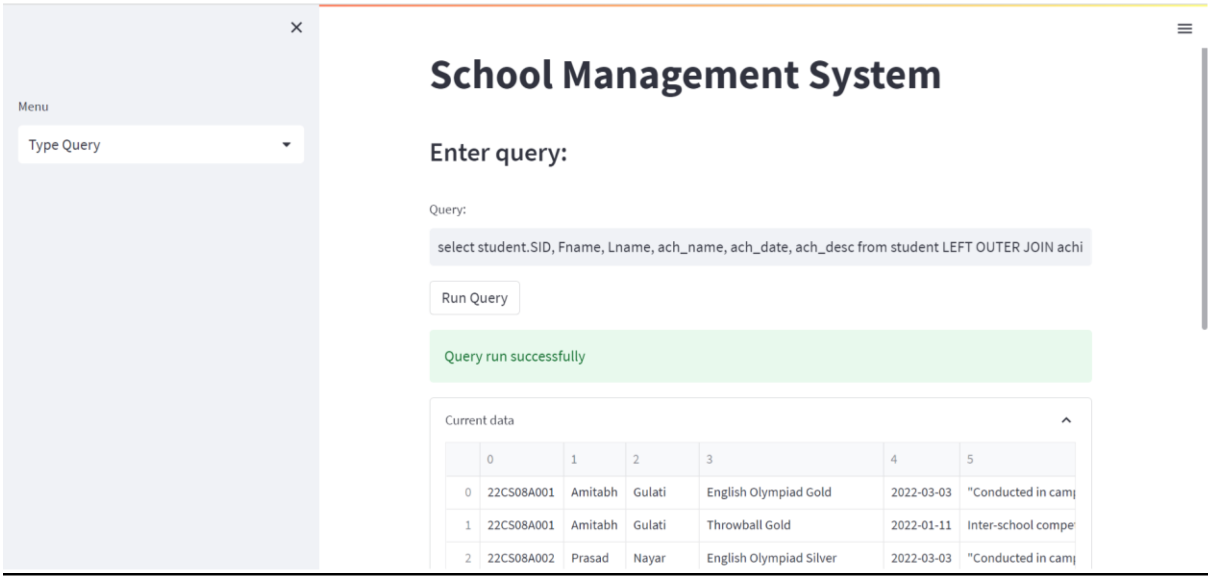
Do you want to delete ::22CS08A009? Student data will be deleted from all tables.

Student has been deleted successfully

Updated data

	SID	Fname	Lname	Age	Gender	Street	City	PIN	Blood_Group
3	22CS08A003	Pratima	Chaudhuri	13	F	Street 5	Bengaluru	560102	B-
4	22CS08B001	Rani	D' Alia	13	F	Street 4	Bengaluru	560051	B+
5	22CS08B002	Nisha	Sanghvi	13	F	Street 3	Bengaluru	560002	A-
6	22CS08B003	Madhuri	Rama	13	F	Street 4	Bengaluru	560051	B-

Type Query:



School Management System

Enter query:

Query:

```
select student.SID, Fname, Lname, ach_name, ach_date, ach_desc from student LEFT OUTER JOIN achi
```

Query run successfully

Current data

	0	1	2	3	4	5
0	22CS08A001	Amitabh	Gulati	English Olympiad Gold	2022-03-03	"Conducted in cam
1	22CS08A001	Amitabh	Gulati	Throwball Gold	2022-01-11	Inter-school compe
2	22CS08A002	Prasad	Nayar	English Olympiad Silver	2022-03-03	"Conducted in cam