DAYANANDA SAGAR UNIVERSITY

School of Engineering

Department Of Computer Science and Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road,
Ramanagara Dt., 562 112

DATABASE MANAGEMENT SYSTEMS

DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DATABASE MANAGEMENT SYSTEMS

22CS2403

NAME:

USN:

Dayananda Sagar University

School of Engineering

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION & MISSION

Vision

- To be recognized as a department of eminence in Computer Science and Engineering focusing on sustainability, inclusive technologies and societal needs.

Mission

- **M1:** Impart quality technical education by designing and delivering contemporary Computer Science Engineering curricula while emphasizing leadership, ethics, values and integrity.
- **M2:** Transform professionals into technically competent through industry-academia collaboration and innovation ecosystem.
- **M3:** Prepare Computer Science and Engineering graduates to meet ever-growing societal needs.

Program Educational Objectives (PEO's)

After few years of graduation, the graduates of Computer Science and Engineering will be able to:

- **PEO1.**Apply appropriate theory, practices and tools in the design, implementation, maintenance and evaluation of computing in the work place or in higher education.
- **PEO2.**Exhibit professional skills in solving challenging problems in their career and advance to leadership roles.
- **PEO3.**Become effective innovator, researcher, and entrepreneur to provide technical solutions for socio-economic challenges.

Programme Outcome (PO's)

Engineering Graduates will be able to:

- **PO1. Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
- **PO2. Problem Analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
- **PO3. Design/Development of Solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
- **PO4. Conduct Investigations of Complex Problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- **PO5. Modern Tool Usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
- **PO6. The Engineer and Society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

- **PO7. Environment and Sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
- **PO8. Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- **PO9. Individual and Team Work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
- **PO10. Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- **PO11. Project Management and Finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- **PO12. Life-long Learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

.

Program Specific Outcomes (PSO's)

Engineering Graduates will be able to:

- **PSO1.**Design and Integrate software and hardware systems by following standard software engineering principles in the areas related to IOT, Cloud, Networks, Security, Embedded Systems, and Artificial Intelligence of varying complexity.
- **PSO2.**Design and Implement application software systems by applying the concepts of Programming languages, Machine Learning, Mobile Computing, and Data Science that meet the automation requirements of society and Industry.

Dayananda Sagar University

SCHOOL OF
ENGINEERING

Laboratory Certificate

This is to certify that Mr./Ms.____bearing University Seat umber (USN)) ______________________________ has satisfactorily completed the ***DATABASE MANAGEMENT SYSTEMS***

prescribed by the University for the______semester, B. Tech.______________branch of this

university during the academic year 20__________20___________

Date: ___________________

____

Signature of the Faculty in charge

Marks	
Maximum	**Obtained**

____

Signature of the Chairman

Instructions to launch MySQL workbench:

- Launch MySQL Workbench to open the home screen. Existing connections are shown when you click the MySQL Connections view from the sidebar. No connections exist for first-time users.
- To add a connection, click the [+] icon to the right of the MySQL Connections title on the home screen. This opens the Setup New Connection. Define the Connection Name value,
- Read the Configure Local Management introduction (shown in the next figure), and press Next to begin defining the new connection parameters.
- he connection will now be tested. You should see that the connection was successful. If not, click Back and check that you have entered the information correctly.

1. Consider the schema for UNIVERSITY Database:

STUDENT(USN, SName, Address, Phone, Gender)

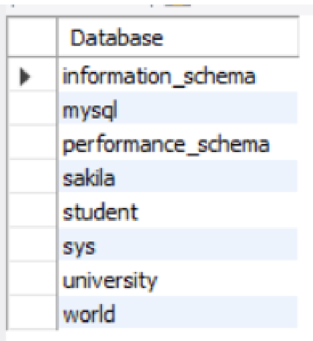
SEMSEC(SSID, Sem, Sec)

CLASS(USN, SSID)

Write SQL queries to:

I. Create table for the above schema, specifying proper Primary Key, Foreign Key. Identify Unique key and Not NULL Attribute.
II. Insert the values for all the tables created.
III. Update student address whose usn= S019.
IV. Delete student details whose usn= S019, and show how it is handled.

CREATE DATABASE UNIVERSITY;
SHOW DATABASES;



Database
information_schema
mysql
performance_schema
sakila
student
sys
university
world

USE UNIVERSITY;

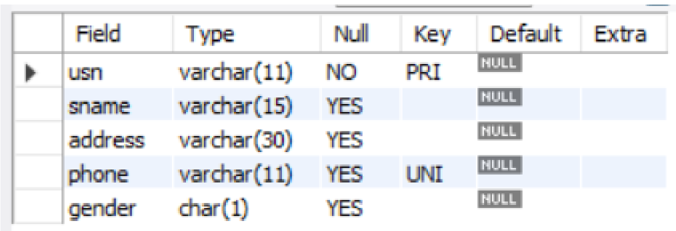
1. Create tables

Student(usn,sname,address,phone,gender)

CREATE TABLE student(usn VARCHAR(11) NOT NULL, sname VARCHAR(15), address VARCHAR(30), phone VARCHAR(11) UNIQUE, gender CHAR, PRIMARY KEY(usn)) ;

OutPut:

desc student;



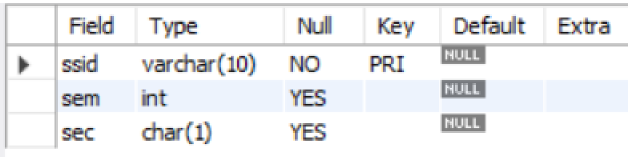
Field	Type	Null	Key	Default	Extra
usn	varchar(11)	NO	PRI	NULL	
sname	varchar(15)	YES		NULL	
address	varchar(30)	YES		NULL	
phone	varchar(11)	YES	UNI	NULL	
gender	char(1)	YES		NULL	

Semsec(ssid,sem,sec)

CREATE TABLE semsec(ssid VARCHAR(10) NOT NULL, sem INT, sec CHAR, PRIMARY KEY(ssid)) ;

Output:

desc semsec;



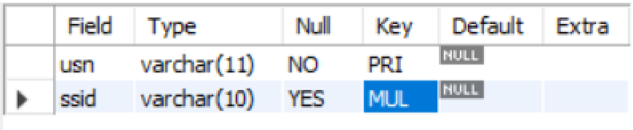
	Field	Type	Null	Key	Default	Extra
▶	ssid	varchar(10)	NO	PRI	NULL	
	sem	int	YES		NULL	
	sec	char(1)	YES		NULL	

Class(usn,ssid)

CREATE TABLE class(usn VARCHAR(11) NOT NULL, ssid VARCHAR(10), PRIMARY KEY(usn), FOREIGN KEY(usn)REFERENCES student(usn) ON DELETE CASCADE, FOREIGN KEY(ssid)REFERENCES semsec(ssid) ON DELETE CASCADE) ;

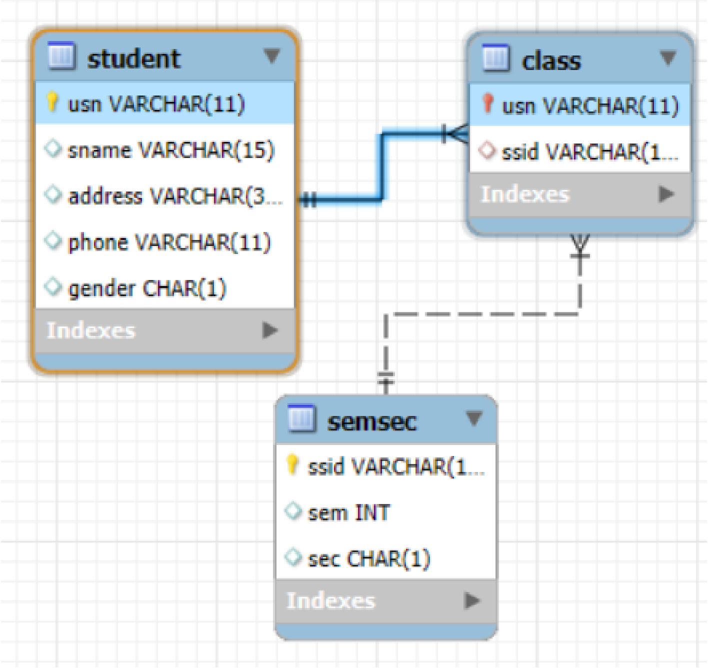
Output:

desc class;



	Field	Type	Null	Key	Default	Extra
	usn	varchar(11)	NO	PRI	NULL	
▶	ssid	varchar(10)	YES	MUL	NULL	

Database→Reverse Engineering



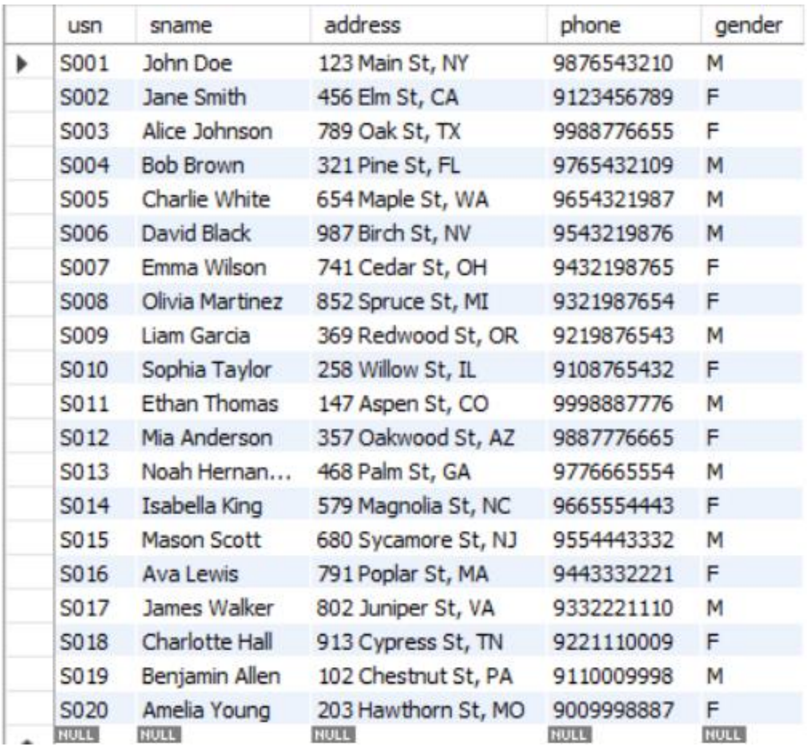
2. Inserting the values for above tables

INSERT INTO Student (USN, SName, Address, Phone, Gender) VALUES ('S001', 'John Doe', '123 Main St, NY', '9876543210', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender) VALUES ('S002', 'Jane Smith', '456 Elm St, CA', '9123456789', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender) VALUES ('S003', 'Alice Johnson', '789 Oak St, TX', '9988776655', 'F');

```
INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S004', 'Bob Brown', '321 Pine St, FL', '9765432109', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S005', 'Charlie White', '654 Maple St, WA', '9654321987', 'M');


INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S006', 'David Black', '987 Birch St, NV', '9543219876', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S007', 'Emma Wilson', '741 Cedar St, OH', '9432198765', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S008', 'Olivia Martinez', '852 Spruce St, MI', '9321987654', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S009', 'Liam Garcia', '369 Redwood St, OR', '9219876543', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S010', 'Sophia Taylor', '258 Willow St, IL', '9108765432', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S011', 'Ethan Thomas', '147 Aspen St, CO', '9998887776', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S012', 'Mia Anderson', '357 Oakwood St, AZ', '9887776665', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S013', 'Noah Hernandez', '468 Palm St, GA', '9776665554', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S014', 'Isabella King', '579 Magnolia St, NC', '9665554443', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S015', 'Mason Scott', '680 Sycamore St, NJ', '9554443332', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S016', 'Ava Lewis', '791 Poplar St, MA', '9443332221', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S017', 'James Walker', '802 Juniper St, VA', '9332221110', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S018', 'Charlotte Hall', '913 Cypress St, TN', '9221110009', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S019', 'Benjamin Allen', '102 Chestnut St, PA', '9110009998', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S020', 'Amelia Young', '203 Hawthorn St, MO', '9009998887', 'F');

Select * from Student;
```



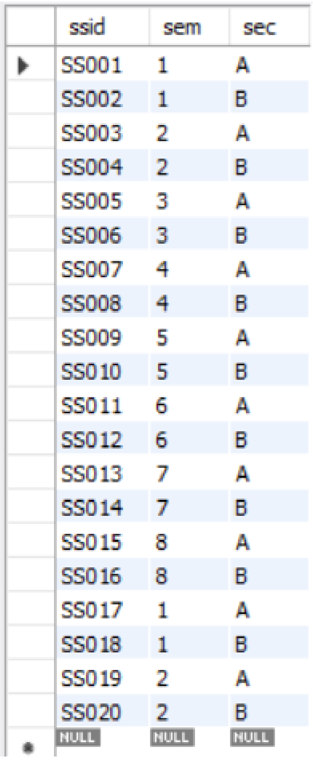
usn	sname	address	phone	gender
S001	John Doe	123 Main St, NY	9876543210	M
S002	Jane Smith	456 Elm St, CA	9123456789	F
S003	Alice Johnson	789 Oak St, TX	9988776655	F
S004	Bob Brown	321 Pine St, FL	9765432109	M
S005	Charlie White	654 Maple St, WA	9654321987	M
S006	David Black	987 Birch St, NV	9543219876	M
S007	Emma Wilson	741 Cedar St, OH	9432198765	F
S008	Olivia Martinez	852 Spruce St, MI	9321987654	F
S009	Liam Garcia	369 Redwood St, OR	9219876543	M
S010	Sophia Taylor	258 Willow St, IL	9108765432	F
S011	Ethan Thomas	147 Aspen St, CO	9998887776	M
S012	Mia Anderson	357 Oakwood St, AZ	9887776665	F
S013	Noah Hernan...	468 Palm St, GA	9776665554	M
S014	Isabella King	579 Magnolia St, NC	9665554443	F
S015	Mason Scott	680 Sycamore St, NJ	9554443332	M
S016	Ava Lewis	791 Poplar St, MA	9443332221	F
S017	James Walker	802 Juniper St, VA	9332221110	M
S018	Charlotte Hall	913 Cypress St, TN	9221110009	F
S019	Benjamin Allen	102 Chestnut St, PA	9110009998	M
S020	Amelia Young	203 Hawthorn St, MO	9009998887	F
NULL	NULL	NULL	NULL	NULL

```
INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS001', 1, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS002', 1, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS003', 2, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS004', 2, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS005', 3, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS006', 3, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS007', 4, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS008', 4, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS009', 5, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS010', 5, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS011', 6, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS012', 6, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS013', 7, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS014', 7, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS015', 8, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS016', 8, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS017', 1, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS018', 1, 'B');
```

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS019', 2, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS020', 2, 'B');

Select * from semsec;



ssid	sem	sec
SS001	1	A
SS002	1	B
SS003	2	A
SS004	2	B
SS005	3	A
SS006	3	B
SS007	4	A
SS008	4	B
SS009	5	A
SS010	5	B
SS011	6	A
SS012	6	B
SS013	7	A
SS014	7	B
SS015	8	A
SS016	8	B
SS017	1	A
SS018	1	B
SS019	2	A
SS020	2	B
NULL	NULL	NULL

INSERT INTO CLASS (USN, SSID) VALUES ('S001', 'SS001');

INSERT INTO CLASS (USN, SSID) VALUES ('S002', 'SS002');

INSERT INTO CLASS (USN, SSID) VALUES ('S003', 'SS003');

INSERT INTO CLASS (USN, SSID) VALUES ('S004', 'SS004');

INSERT INTO CLASS (USN, SSID) VALUES ('S005', 'SS005');

INSERT INTO CLASS (USN, SSID) VALUES ('S006', 'SS006');

INSERT INTO CLASS (USN, SSID) VALUES ('S007', 'SS007');

INSERT INTO CLASS (USN, SSID) VALUES ('S008', 'SS008');

INSERT INTO CLASS (USN, SSID) VALUES ('S009', 'SS009');

INSERT INTO CLASS (USN, SSID) VALUES ('S010', 'SS010');

INSERT INTO CLASS (USN, SSID) VALUES ('S011', 'SS011');

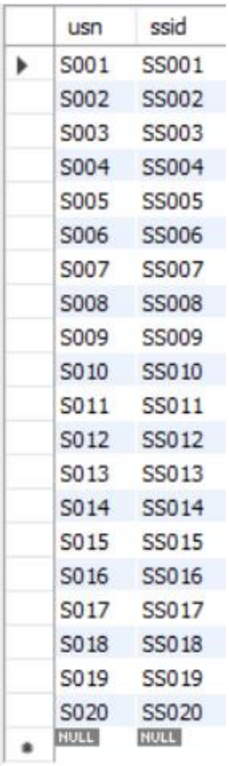
INSERT INTO CLASS (USN, SSID) VALUES ('S012', 'SS012');

INSERT INTO CLASS (USN, SSID) VALUES ('S013', 'SS013');

INSERT INTO CLASS (USN, SSID) VALUES ('S014', 'SS014');

INSERT INTO CLASS (USN, SSID) VALUES ('S015', 'SS015');

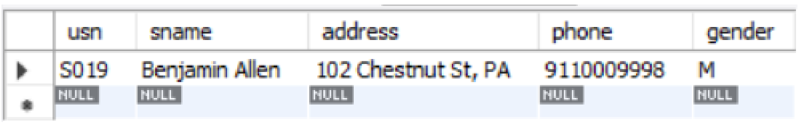
```
INSERT INTO CLASS (USN, SSID) VALUES ('S016', 'SS016');

INSERT INTO CLASS (USN, SSID) VALUES ('S017', 'SS017');

INSERT INTO CLASS (USN, SSID) VALUES ('S018', 'SS018');

INSERT INTO CLASS (USN, SSID) VALUES ('S019', 'SS019');

INSERT INTO CLASS (USN, SSID) VALUES ('S020', 'SS020');

Select * from class;
```



usn	ssid
S001	SS001
S002	SS002
S003	SS003
S004	SS004
S005	SS005
S006	SS006
S007	SS007
S008	SS008
S009	SS009
S010	SS010
S011	SS011
S012	SS012
S013	SS013
S014	SS014
S015	SS015
S016	SS016
S017	SS017
S018	SS018
S019	SS019
S020	SS020
NULL	NULL

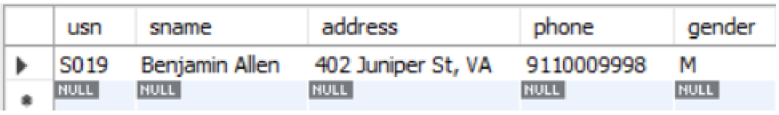
3. Update address of student with usn s019

```
select * from student where usn= 'S019';
```



usn	sname	address	phone	gender
S019	Benjamin Allen	102 Chestnut St, PA	9110009998	M
NULL	NULL	NULL	NULL	NULL

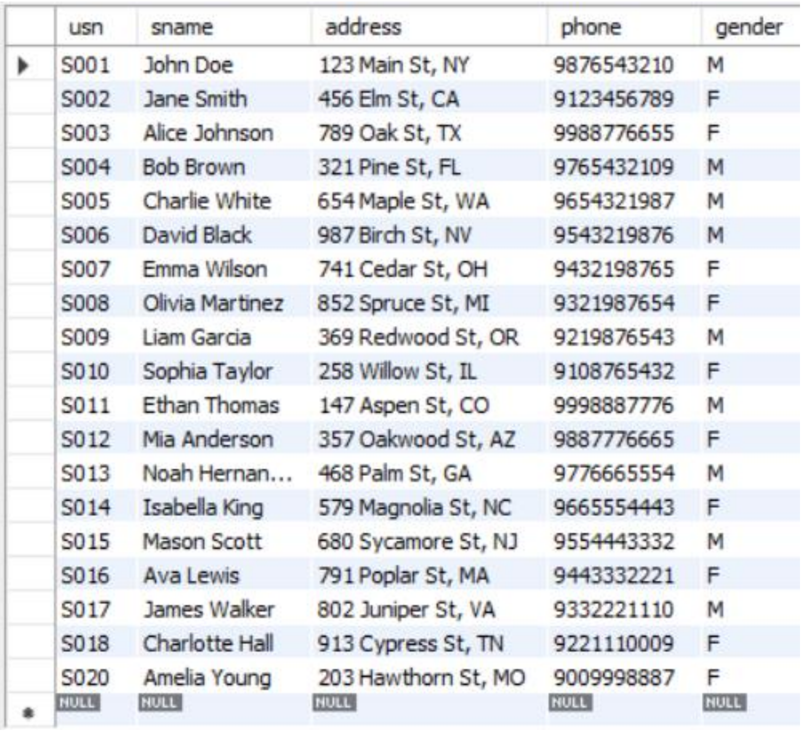
```
UPDATE STUDENT SET Address='402 Juniper St, VA' WHERE usn='S019';

select * from student where usn= 'S019';
```



usn	sname	address	phone	gender
S019	Benjamin Allen	402 Juniper St, VA	9110009998	M
NULL	NULL	NULL	NULL	NULL

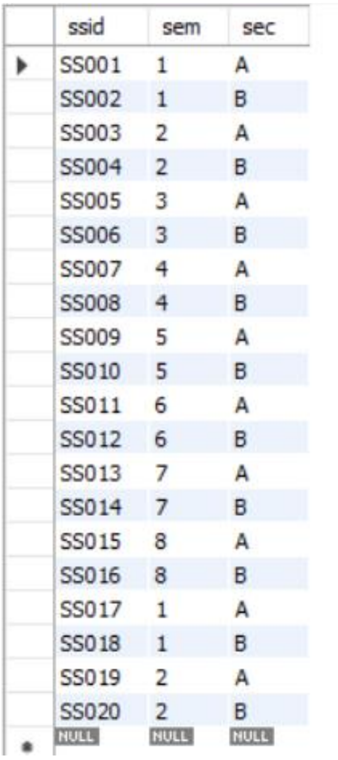
4. Delete student details whose usn= S019

```
delete from student where usn= 'S019';

SELECT * FROM STUDENT;
```



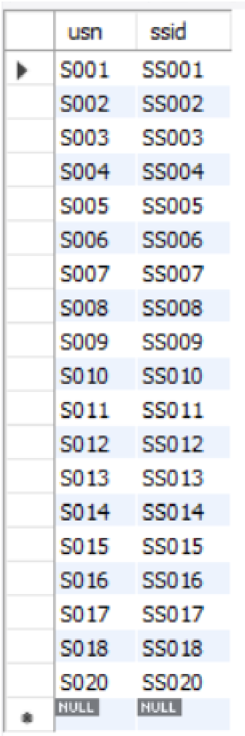
usn	sname	address	phone	gender
S001	John Doe	123 Main St, NY	9876543210	M
S002	Jane Smith	456 Elm St, CA	9123456789	F
S003	Alice Johnson	789 Oak St, TX	9988776655	F
S004	Bob Brown	321 Pine St, FL	9765432109	M
S005	Charlie White	654 Maple St, WA	9654321987	M
S006	David Black	987 Birch St, NV	9543219876	M
S007	Emma Wilson	741 Cedar St, OH	9432198765	F
S008	Olivia Martinez	852 Spruce St, MI	9321987654	F
S009	Liam Garcia	369 Redwood St, OR	9219876543	M
S010	Sophia Taylor	258 Willow St, IL	9108765432	F
S011	Ethan Thomas	147 Aspen St, CO	9998887776	M
S012	Mia Anderson	357 Oakwood St, AZ	9887776665	F
S013	Noah Hernan...	468 Palm St, GA	9776665554	M
S014	Isabella King	579 Magnolia St, NC	9665554443	F
S015	Mason Scott	680 Sycamore St, NJ	9554443332	M
S016	Ava Lewis	791 Poplar St, MA	9443332221	F
S017	James Walker	802 Juniper St, VA	9332221110	M
S018	Charlotte Hall	913 Cypress St, TN	9221110009	F
S020	Amelia Young	203 Hawthorn St, MO	9009998887	F
NULL	NULL	NULL	NULL	NULL

SELECT * FROM SEMSEC;



ssid	sem	sec
SS001	1	A
SS002	1	B
SS003	2	A
SS004	2	B
SS005	3	A
SS006	3	B
SS007	4	A
SS008	4	B
SS009	5	A
SS010	5	B
SS011	6	A
SS012	6	B
SS013	7	A
SS014	7	B
SS015	8	A
SS016	8	B
SS017	1	A
SS018	1	B
SS019	2	A
SS020	2	B
NULL	NULL	NULL

SELECT * FROM CLASS;



usn	ssid
S001	SS001
S002	SS002
S003	SS003
S004	SS004
S005	SS005
S006	SS006
S007	SS007
S008	SS008
S009	SS009
S010	SS010
S011	SS011
S012	SS012
S013	SS013
S014	SS014
S015	SS015
S016	SS016
S017	SS017
S018	SS018
S020	SS020
NULL	NULL

2. Consider the schema for UNIVERSITY Database:

STUDENT(USN, SName, Address, Phone, Gender)

SEMSEC(SSID, Sem, Sec)

CLASS(USN, SSID)

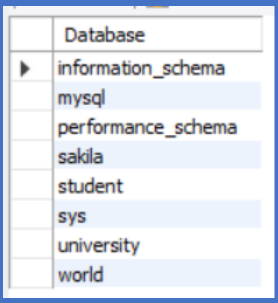
SUBJECT(Subcode, Title, Sem, Credits)

IAMARKS(USN, Subcode, SSID, Test1, Test2, Test3, FinalIA)

Write SQL queries to:

I. Create table for the above schema, specifying proper Primary Key, Foreign Key. Identify Unique key and Not NULL Attribute.
II. Insert the values for all the tables created.
III. Retive the details of student having usn='S018' from all the tables using Referential integrity.
IV. Retrieve the top 3 male students who have scored in DBMS subject for the Test1, and display their details like usn, name, gender and marks. Sort the students based on marks from high to low.

```
CREATE DATABASE UNIVERSITY;
SHOW DATABASES;
```



Database
information_schema
mysql
performance_schema
sakila
student
sys
university
world

```
USE UNIVERSITY;
```

Drop database student;

"If you need to drop previous and create New of your own."

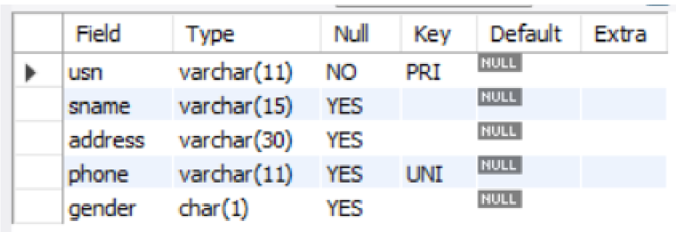
1. Create tables

Student(usn,sname,address,phone,gender)

CREATE TABLE student(usn VARCHAR(11) NOT NULL, sname VARCHAR(15), address VARCHAR(30), phone VARCHAR(11) UNIQUE, gender CHAR, PRIMARY KEY(usn)) ;

OutPut:

desc student;



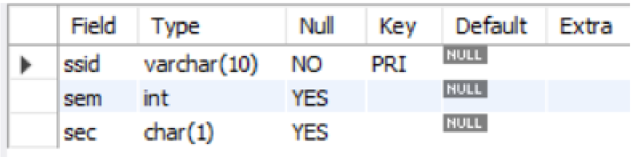
	Field	Type	Null	Key	Default	Extra
▶	usn	varchar(11)	NO	PRI	NULL	
	sname	varchar(15)	YES		NULL	
	address	varchar(30)	YES		NULL	
	phone	varchar(11)	YES	UNI	NULL	
	gender	char(1)	YES		NULL	

Semsec(ssid,sem,sec)

CREATE TABLE semsec(ssid VARCHAR(10) NOT NULL, sem INT, sec CHAR, PRIMARY KEY(ssid)) ;

Output:

desc semsec;



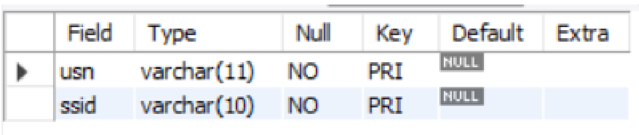
	Field	Type	Null	Key	Default	Extra
▶	ssid	varchar(10)	NO	PRI	NULL	
	sem	int	YES		NULL	
	sec	char(1)	YES		NULL	

Class(usn,ssid)

CREATE TABLE class(usn VARCHAR(11) NOT NULL, ssid VARCHAR(10), PRIMARY KEY(usn,ssid), FOREIGN KEY(usn)REFERENCES student(usn) ON DELETE CASCADE, FOREIGN KEY(ssid)REFERENCES semsec(ssid) ON DELETE CASCADE) ;

Output:

desc class;



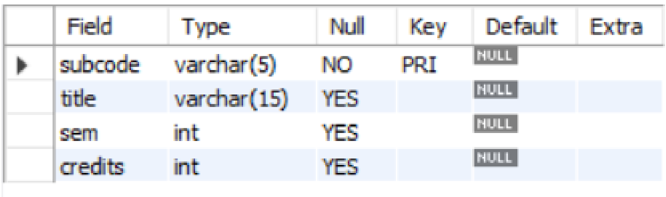
	Field	Type	Null	Key	Default	Extra
▶	usn	varchar(11)	NO	PRI	NULL	
	ssid	varchar(10)	NO	PRI	NULL	

SUBJECT(Subcode, Title, Sem, Credits)

CREATE TABLE sub(subcode VARCHAR(5) NOT NULL, title VARCHAR(15), sem INT, credits INT, PRIMARY KEY(subcode));

Output:

desc sub;



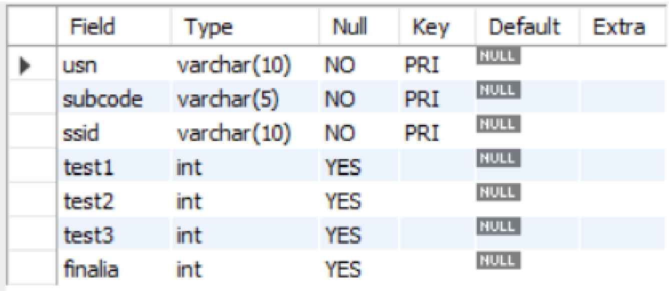
Field	Type	Null	Key	Default	Extra
subcode	varchar(5)	NO	PRI	NULL	
title	varchar(15)	YES		NULL	
sem	int	YES		NULL	
credits	int	YES		NULL	

Iamarks(usn,subcode,ssid,test1,test2,test3,finalia)

CREATE TABLE iamarks(usn VARCHAR(10) NOT NULL, subcode VARCHAR(5), ssid VARCHAR(10), test1 INT, test2 INT, test3 INT, finalia INT, PRIMARY KEY(usn,subcode,ssid), FOREIGN KEY(usn)REFERENCES student(usn) ON DELETE CASCADE, FOREIGN KEY(subcode)REFERENCES sub(subcode) ON DELETE CASCADE, FOREIGN KEY(ssid)REFERENCES semsec(ssid) ON DELETE CASCADE);

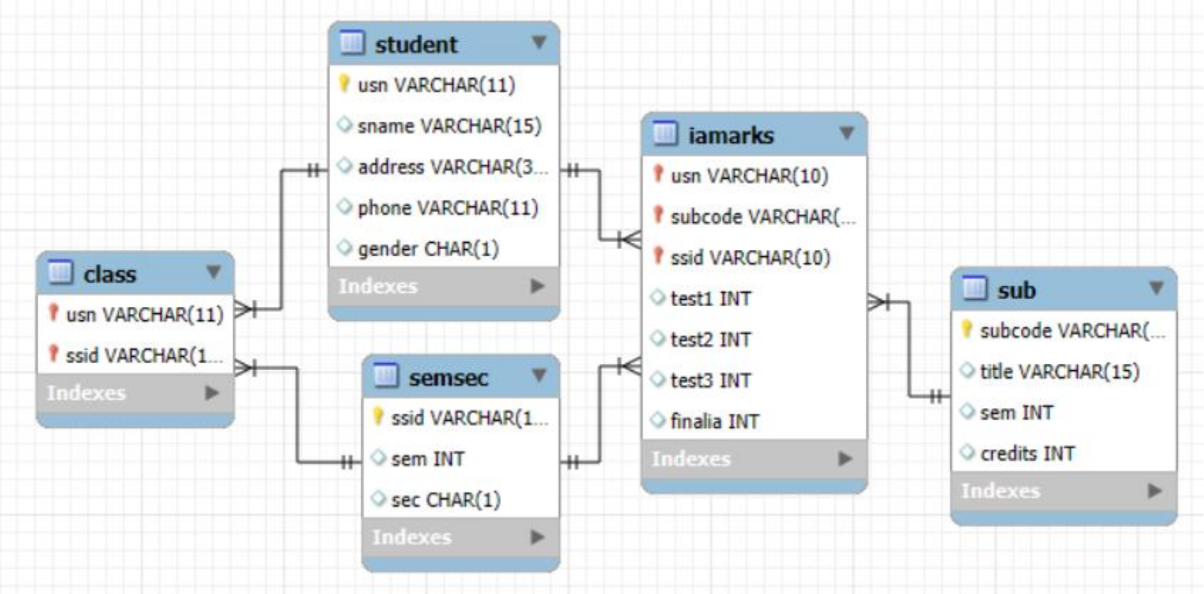
Output:

desc iamarks;



Field	Type	Null	Key	Default	Extra
usn	varchar(10)	NO	PRI	NULL	
subcode	varchar(5)	NO	PRI	NULL	
ssid	varchar(10)	NO	PRI	NULL	
test1	int	YES		NULL	
test2	int	YES		NULL	
test3	int	YES		NULL	
finalia	int	YES		NULL	

Database→Reverse Engineering

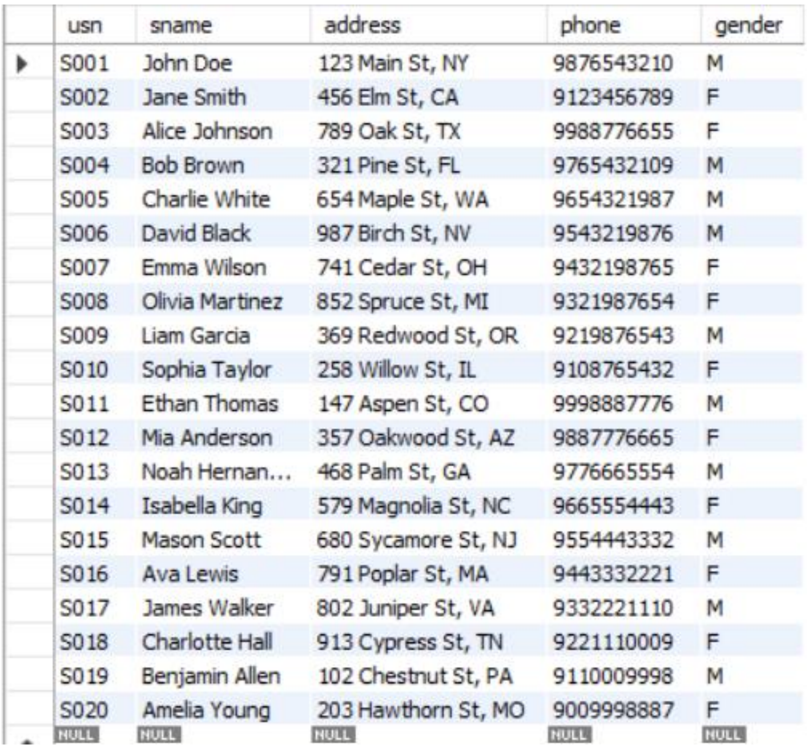


2. Inserting the values for above tables

INSERT INTO Student (USN, SName, Address, Phone, Gender) VALUES ('S001', 'John Doe', '123 Main St, NY', '9876543210', 'M');

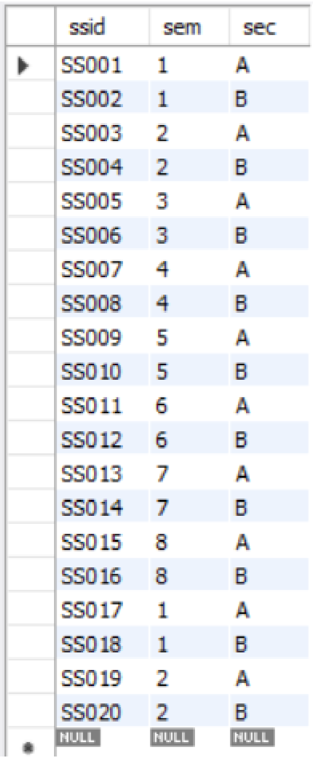
INSERT INTO Student (USN, SName, Address, Phone, Gender) VALUES ('S002', 'Jane Smith', '456 Elm St, CA', '9123456789', 'F');

```sql
INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S003', 'Alice Johnson', '789 Oak St, TX', '9988776655', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S004', 'Bob Brown', '321 Pine St, FL', '9765432109', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S005', 'Charlie White', '654 Maple St, WA', '9654321987', 'M');


INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S006', 'David Black', '987 Birch St, NV', '9543219876', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S007', 'Emma Wilson', '741 Cedar St, OH', '9432198765', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S008', 'Olivia Martinez', '852 Spruce St, MI', '9321987654', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S009', 'Liam Garcia', '369 Redwood St, OR', '9219876543', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S010', 'Sophia Taylor', '258 Willow St, IL', '9108765432', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S011', 'Ethan Thomas', '147 Aspen St, CO', '9998887776', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S012', 'Mia Anderson', '357 Oakwood St, AZ', '9887776665', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S013', 'Noah Hernandez', '468 Palm St, GA', '9776665554', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S014', 'Isabella King', '579 Magnolia St, NC', '9665554443', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S015', 'Mason Scott', '680 Sycamore St, NJ', '9554443332', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S016', 'Ava Lewis', '791 Poplar St, MA', '9443332221', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S017', 'James Walker', '802 Juniper St, VA', '9332221110', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S018', 'Charlotte Hall', '913 Cypress St, TN', '9221110009', 'F');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S019', 'Benjamin Allen', '102 Chestnut St, PA', '9110009998', 'M');

INSERT INTO Student (USN, SName, Address, Phone, Gender)  VALUES ('S020', 'Amelia Young', '203 Hawthorn St, MO', '9009998887', 'F');

Select * from Student;
```



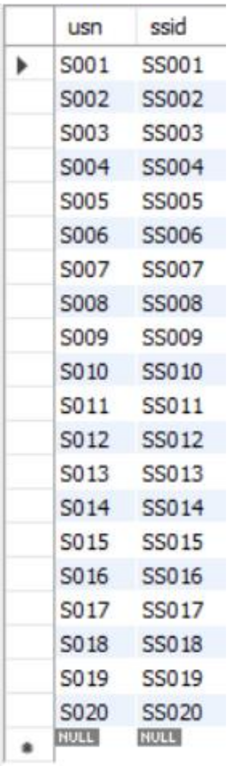
usn	sname	address	phone	gender
S001	John Doe	123 Main St, NY	9876543210	M
S002	Jane Smith	456 Elm St, CA	9123456789	F
S003	Alice Johnson	789 Oak St, TX	9988776655	F
S004	Bob Brown	321 Pine St, FL	9765432109	M
S005	Charlie White	654 Maple St, WA	9654321987	M
S006	David Black	987 Birch St, NV	9543219876	M
S007	Emma Wilson	741 Cedar St, OH	9432198765	F
S008	Olivia Martinez	852 Spruce St, MI	9321987654	F
S009	Liam Garcia	369 Redwood St, OR	9219876543	M
S010	Sophia Taylor	258 Willow St, IL	9108765432	F
S011	Ethan Thomas	147 Aspen St, CO	9998887776	M
S012	Mia Anderson	357 Oakwood St, AZ	9887776665	F
S013	Noah Hernan...	468 Palm St, GA	9776665554	M
S014	Isabella King	579 Magnolia St, NC	9665554443	F
S015	Mason Scott	680 Sycamore St, NJ	9554443332	M
S016	Ava Lewis	791 Poplar St, MA	9443332221	F
S017	James Walker	802 Juniper St, VA	9332221110	M
S018	Charlotte Hall	913 Cypress St, TN	9221110009	F
S019	Benjamin Allen	102 Chestnut St, PA	9110009998	M
S020	Amelia Young	203 Hawthorn St, MO	9009998887	F
NULL	NULL	NULL	NULL	NULL

```
INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS001', 1, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS002', 1, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS003', 2, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS004', 2, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS005', 3, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS006', 3, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS007', 4, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS008', 4, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS009', 5, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS010', 5, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS011', 6, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS012', 6, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS013', 7, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS014', 7, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS015', 8, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS016', 8, 'B');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS017', 1, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS018', 1, 'B');
INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS019', 2, 'A');

INSERT INTO SEMSEC (SSID, Sem, Sec) VALUES ('SS020', 2, 'B');

Select * from semsec;
```



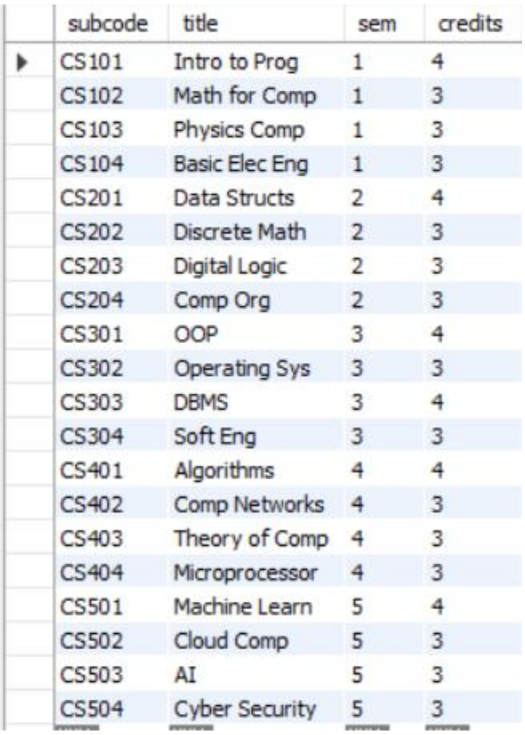
ssid	sem	sec
SS001	1	A
SS002	1	B
SS003	2	A
SS004	2	B
SS005	3	A
SS006	3	B
SS007	4	A
SS008	4	B
SS009	5	A
SS010	5	B
SS011	6	A
SS012	6	B
SS013	7	A
SS014	7	B
SS015	8	A
SS016	8	B
SS017	1	A
SS018	1	B
SS019	2	A
SS020	2	B
NULL	NULL	NULL

```
INSERT INTO CLASS (USN, SSID) VALUES ('S001', 'SS001');

INSERT INTO CLASS (USN, SSID) VALUES ('S002', 'SS002');

INSERT INTO CLASS (USN, SSID) VALUES ('S003', 'SS003');

INSERT INTO CLASS (USN, SSID) VALUES ('S004', 'SS004');

INSERT INTO CLASS (USN, SSID) VALUES ('S005', 'SS005');

INSERT INTO CLASS (USN, SSID) VALUES ('S006', 'SS006');

INSERT INTO CLASS (USN, SSID) VALUES ('S007', 'SS007');

INSERT INTO CLASS (USN, SSID) VALUES ('S008', 'SS008');

INSERT INTO CLASS (USN, SSID) VALUES ('S009', 'SS009');

INSERT INTO CLASS (USN, SSID) VALUES ('S010', 'SS010');

INSERT INTO CLASS (USN, SSID) VALUES ('S011', 'SS011');

INSERT INTO CLASS (USN, SSID) VALUES ('S012', 'SS012');

INSERT INTO CLASS (USN, SSID) VALUES ('S013', 'SS013');

INSERT INTO CLASS (USN, SSID) VALUES ('S014', 'SS014');

INSERT INTO CLASS (USN, SSID) VALUES ('S015', 'SS015');
INSERT INTO CLASS (USN, SSID) VALUES ('S016', 'SS016');

INSERT INTO CLASS (USN, SSID) VALUES ('S017', 'SS017');

INSERT INTO CLASS (USN, SSID) VALUES ('S018', 'SS018');

INSERT INTO CLASS (USN, SSID) VALUES ('S019', 'SS019');

INSERT INTO CLASS (USN, SSID) VALUES ('S020', 'SS020');

Select * from class;
```



usn	ssid
S001	SS001
S002	SS002
S003	SS003
S004	SS004
S005	SS005
S006	SS006
S007	SS007
S008	SS008
S009	SS009
S010	SS010
S011	SS011
S012	SS012
S013	SS013
S014	SS014
S015	SS015
S016	SS016
S017	SS017
S018	SS018
S019	SS019
S020	SS020
NULL	NULL

```
INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS101', 'Intro to Prog', 1, 4);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS102', 'Math for Comp', 1, 3);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS103', 'Physics Comp', 1, 3);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS104', 'Basic Elec Eng', 1, 3);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS201', 'Data Structs', 2, 4);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS202', 'Discrete Math', 2, 3);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS203', 'Digital Logic', 2, 3);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS204', 'Comp Org', 2, 3);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS301', 'OOP', 3, 4);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS302', 'Operating Sys', 3, 3);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS303', 'DBMS', 3, 4);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS304', 'Soft Eng', 3, 3);
INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS401', 'Algorithms', 4, 4);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS402', 'Comp Networks', 4, 3);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS403', 'Theory of Comp', 4, 3);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS404', 'Microprocessor', 4, 3);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS501', 'Machine Learn', 5, 4);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS502', 'Cloud Comp', 5, 3);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS503', 'AI', 5, 3);

INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS504', 'Cyber Security', 5, 3);

Select * from sub;
```

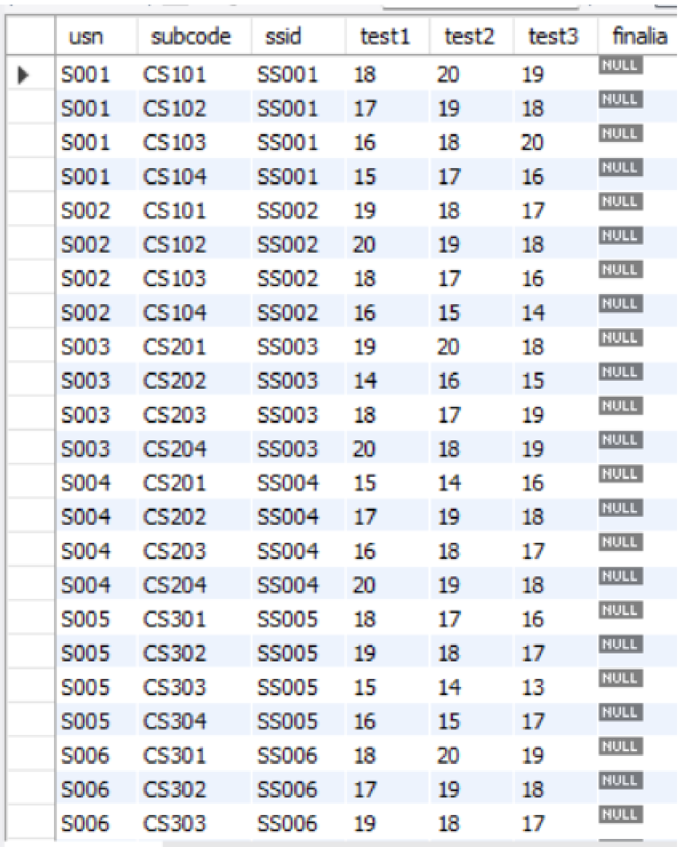


subcode	title	sem	credits
CS101	Intro to Prog	1	4
CS102	Math for Comp	1	3
CS103	Physics Comp	1	3
CS104	Basic Elec Eng	1	3
CS201	Data Structs	2	4
CS202	Discrete Math	2	3
CS203	Digital Logic	2	3
CS204	Comp Org	2	3
CS301	OOP	3	4
CS302	Operating Sys	3	3
CS303	DBMS	3	4
CS304	Soft Eng	3	3
CS401	Algorithms	4	4
CS402	Comp Networks	4	3
CS403	Theory of Comp	4	3
CS404	Microprocessor	4	3
CS501	Machine Learn	5	4
CS502	Cloud Comp	5	3
CS503	AI	5	3
CS504	Cyber Security	5	3

```
INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S001', 'CS101', 'SS001', 18, 20, 19);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S001', 'CS102', 'SS001', 17, 19, 18);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S001', 'CS103', 'SS001', 16, 18, 20);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S001', 'CS104', 'SS001', 15, 17, 16);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S002', 'CS101', 'SS002', 19, 18, 17);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S002', 'CS102', 'SS002', 20, 19, 18);
```

```sql
INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S002', 'CS103', 'SS002', 18, 17, 16);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S002', 'CS104', 'SS002', 16, 15, 14);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S003', 'CS201', 'SS003', 19, 20, 18);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S003', 'CS202', 'SS003', 14, 16, 15);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S003', 'CS203', 'SS003', 18, 17, 19);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S003', 'CS204', 'SS003', 20, 18, 19);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S004', 'CS201', 'SS004', 15, 14, 16);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S004', 'CS202', 'SS004', 17, 19, 18);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S004', 'CS203', 'SS004', 16, 18, 17);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S004', 'CS204', 'SS004', 20, 19, 18);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S005', 'CS301', 'SS005', 18, 17, 16);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S005', 'CS302', 'SS005', 19, 18, 17);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S005', 'CS303', 'SS005', 15, 14, 13);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S005', 'CS304', 'SS005', 16, 15, 17);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S006', 'CS301', 'SS006', 18, 20, 19);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S006', 'CS302', 'SS006', 17, 19, 18);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S006', 'CS303', 'SS006', 19, 18, 17);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S006', 'CS304', 'SS006', 20, 19, 18);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S007', 'CS401', 'SS007', 18, 17, 16);
INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S007', 'CS402', 'SS007', 19, 18, 17);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S007', 'CS403', 'SS007', 15, 14, 13);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S007', 'CS404', 'SS007', 16, 15, 17);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S008', 'CS401', 'SS008', 18, 20, 19);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S008', 'CS402', 'SS008', 17, 19, 18);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S008', 'CS403', 'SS008', 19, 18, 17);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S008', 'CS404', 'SS008', 20, 19, 18);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S009', 'CS501', 'SS009', 18, 17, 16);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S009', 'CS502', 'SS009', 19, 18, 17);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S009', 'CS503', 'SS009', 15, 14, 13);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S009', 'CS504', 'SS009', 16, 15, 17);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S010', 'CS501', 'SS010', 18, 20, 19);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S010', 'CS502', 'SS010', 17, 19, 18);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S010', 'CS503', 'SS010', 19, 18, 17);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S010', 'CS504', 'SS010', 20, 19, 18);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S011', 'CS101', 'SS011', 20, 18, 19);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S011', 'CS102', 'SS011', 18, 17, 16);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S012', 'CS103', 'SS012', 17, 19, 18);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S012', 'CS104', 'SS012', 20, 19, 18);
```

```sql
INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S013', 'CS201', 'SS013', 16, 18, 17);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S014', 'CS202', 'SS014', 15, 17, 16);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S014', 'CS203', 'SS014', 18, 19, 20);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S015', 'CS204', 'SS015', 14, 15, 13);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S016', 'CS301', 'SS016', 19, 18, 17);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S016', 'CS302', 'SS016', 16, 15, 14);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S017', 'CS303', 'SS017', 20, 19, 18);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S018', 'CS304', 'SS018', 18, 20, 19);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S018', 'CS401', 'SS018', 17, 18, 16);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S019', 'CS402', 'SS019', 16, 17, 18);

INSERT INTO IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3) VALUES ('S020', 'CS403', 'SS020', 14, 16, 15);
```



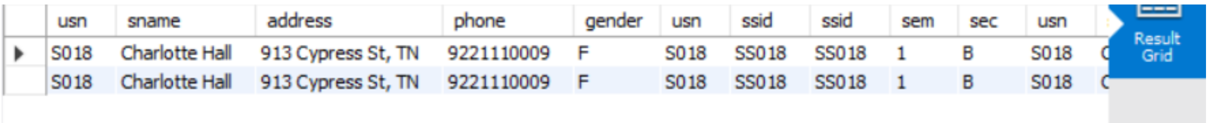
usn	subcode	ssid	test1	test2	test3	finalia
S001	CS101	SS001	18	20	19	NULL
S001	CS102	SS001	17	19	18	NULL
S001	CS103	SS001	16	18	20	NULL
S001	CS104	SS001	15	17	16	NULL
S002	CS101	SS002	19	18	17	NULL
S002	CS102	SS002	20	19	18	NULL
S002	CS103	SS002	18	17	16	NULL
S002	CS104	SS002	16	15	14	NULL
S003	CS201	SS003	19	20	18	NULL
S003	CS202	SS003	14	16	15	NULL
S003	CS203	SS003	18	17	19	NULL
S003	CS204	SS003	20	18	19	NULL
S004	CS201	SS004	15	14	16	NULL
S004	CS202	SS004	17	19	18	NULL
S004	CS203	SS004	16	18	17	NULL
S004	CS204	SS004	20	19	18	NULL
S005	CS301	SS005	18	17	16	NULL
S005	CS302	SS005	19	18	17	NULL
S005	CS303	SS005	15	14	13	NULL
S005	CS304	SS005	16	15	17	NULL
S006	CS301	SS006	18	20	19	NULL
S006	CS302	SS006	17	19	18	NULL
S006	CS303	SS006	19	18	17	NULL

3. Retive the details of student having usn=’S018’ from all the tables using Referential integrity.

SELECT *

FROM STUDENT S , CLASS C, SEMSEC SS, IAMARKS IA, SUB

WHERE S.USN = C.USN AND C.SSID = SS.SSID AND S.USN = IA.USN AND IA.Subcode = SUB.Subcode AND S.USN ='S018';



usn	sname	address	phone	gender	usn	ssid	ssid	sem	sec	usn	
S018	Charlotte Hall	913 Cypress St, TN	9221110009	F	S018	SS018	SS018	1	B	S018	C
S018	Charlotte Hall	913 Cypress St, TN	9221110009	F	S018	SS018	SS018	1	B	S018	C

4. Retrieve the top 3 male students who have scored in DBMS subject for the Test1, and display their details like usn, name, gender and marks. Sort the students based on marks from high to low.

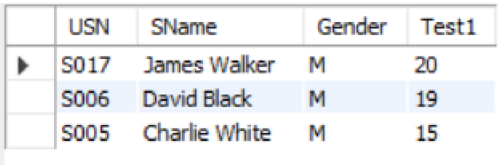
SELECT IA.USN, S.SName, S.Gender, IA.Test1

FROM IAMARKS IA, STUDENT S , SUB

WHERE IA.USN = S.USN AND IA.Subcode = SUB.Subcode AND S.Gender = 'M' AND SUB.Title = 'DBMS'

ORDER BY IA.Test1 DESC

LIMIT 3;



	USN	SName	Gender	Test1
▶	S017	James Walker	M	20
	S006	David Black	M	19
	S005	Charlie White	M	15

3. Write SQL queries to:
   1. Identify 5 Fast learners considering performance of both Test 1 & Test2 combined, Modify the query to retrieve 5 slow learners.
   2. Calculate the FinalIA (average of best of two test marks) and update the corresponding table for all students.
   3. Categorize students based on the following criterion If finalia=17 to 20 then cat='Outstanding' If finalia=12 to 16 then cat='Average' If finalia<12 then cat='weak' Give these details only for 8th semester A,B, and C section students.

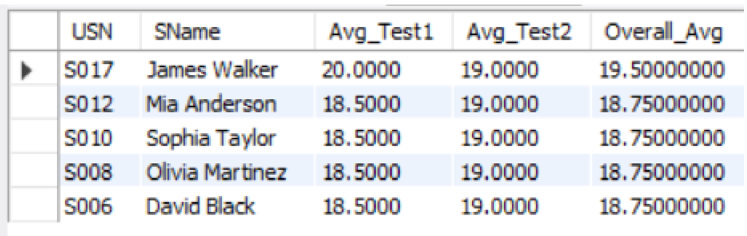
1. Identify 5 Fast learners considering performance of both Test 1 & Test2 combined, Modify the query to retrieve 5 slow learners.

```
SELECT IA.USN, S.SName,

    AVG(IA.Test1) AS Avg_Test1,

    AVG(IA.Test2) AS Avg_Test2,

    (AVG(IA.Test1) + AVG(IA.Test2)) / 2 AS Overall_Avg

FROM IAMARKS IA

JOIN STUDENT S ON IA.USN = S.USN

GROUP BY IA.USN, S.SName

ORDER BY Overall_Avg DESC

LIMIT 5;
```

How will it get slow learners…?

DESC → ASC

Let me check…!!!



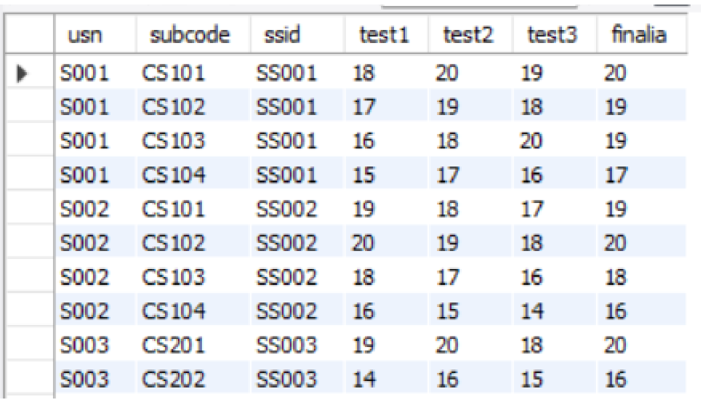
	USN	SName	Avg_Test1	Avg_Test2	Overall_Avg
▶	S017	James Walker	20.0000	19.0000	19.50000000
	S012	Mia Anderson	18.5000	19.0000	18.75000000
	S010	Sophia Taylor	18.5000	19.0000	18.75000000
	S008	Olivia Martinez	18.5000	19.0000	18.75000000
	S006	David Black	18.5000	19.0000	18.75000000

2. Calculate the FinalIA, Such that average of best of two test marks among three test marks should be considered and update the corresponding table for all students.

```
UPDATE iamarks SET finalia=

CASE WHEN test1<test2 AND test1<test3 THEN (test2+ test3)/2

WHEN test2<test1 AND test2<test3 THEN (test1+ test3)/2

ELSE (test1+ test2)/2

END
```

It's not Working for you..?

Well follow this hack to disable Safe Updates,

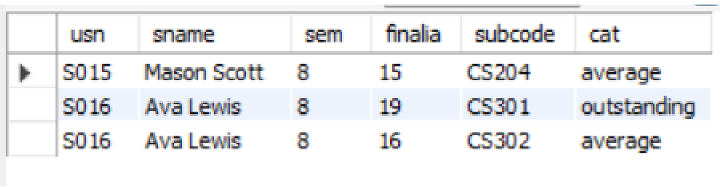


	usn	subcode	ssid	test1	test2	test3	finalia
▶	S001	CS101	SS001	18	20	19	20
	S001	CS102	SS001	17	19	18	19
	S001	CS103	SS001	16	18	20	19
	S001	CS104	SS001	15	17	16	17
	S002	CS101	SS002	19	18	17	19
	S002	CS102	SS002	20	19	18	20
	S002	CS103	SS002	18	17	16	18
	S002	CS104	SS002	16	15	14	16
	S003	CS201	SS003	19	20	18	20
	S003	CS202	SS003	14	16	15	16

3. Categorize students based on the following criterion If finalia=17 to 20 then cat=’Outstanding’ If finalia=12 to 16 then cat=’Average’ If finalia<12 then cat=’weak’ Give these details only for 8th semester A,B, and C section students.

```
ALTER TABLE iamarks ADD COLUMN cat VARCHAR(30)

UPDATE iamarks SET cat= CASE WHEN finalia<=20 AND finalia>=17 THEN 'outstanding' WHEN
finalia<=16 AND finalia>=12 THEN 'average' ELSE 'weak' END WHERE ssid IN (SELECT ssid FROM
semsec WHERE sem=8 AND sec='a' OR sem=8 AND sec='b' OR sem=8 AND sec='c');
```

```
SELECT s.usn,s.sname,ss.sem,ia.finalia,ia.subcode,ia.cat

FROM STUDENT S , CLASS C, SEMSEC SS, IAMARKS IA

WHERE S.USN = C.USN AND C.SSID = SS.SSID  AND S.USN = IA.USN and ss.sem=8;
```

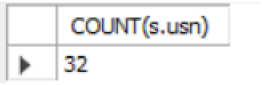


	usn	sname	sem	finalia	subcode	cat
▶	S015	Mason Scott	8	15	CS204	average
	S016	Ava Lewis	8	19	CS301	outstanding
	S016	Ava Lewis	8	16	CS302	average

4. Write SQL queries to:
    1. Count the number of students whose final IA marks is above SUBJECT average FOR DBMS.
    2. Find the total count of Credits for the given student 'ava lewis'.
    3. Classify all FEMALE and MALE students who have opted for 'Digital Logic' using UNION.

1. Count the number of students whose final IA marks is above SUBJECT average FOR DBMS.

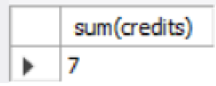
```
SELECT COUNT(s.usn) FROM student s, IAMARKS IA

WHERE IA.usn=s.usn AND finalia>

(SELECT AVG(finalia) FROM iamarks ia

WHERE ia.subcode=(select subcode from sub where title='DBMS'));
```



	COUNT(s.usn)
▶	32

2. Find the total count of Credits for the given student ‘Mason scott’

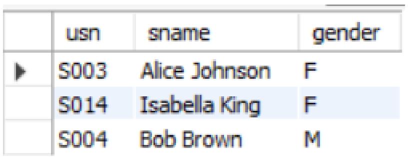
```
SELECT sum(credits)

from sub s, iamarks ia, class c, student st

where s.subcode=ia.subcode and

ia.ssid=c.ssid and

c.usn=st.usn and st.usn IN (select usn from student where sname=' ava lewis');
```



	sum(credits)
▶	7

3. Classify all FEMALE and MALE students who have opted for 'Digital Logic' using UNION.

```
(SELECT st.usn,st.sname,st.gender

from sub s, iamarks ia, student st

where s.subcode=ia.subcode and

ia.usn=st.usn and s.title='Digital Logic' AND st.gender='f')

UNION

(SELECT st.usn,st.sname,st.gender

from sub s, iamarks ia, student st

where s.subcode=ia.subcode and

ia.usn=st.usn and s.title='Digital Logic' AND st.gender='m');
```

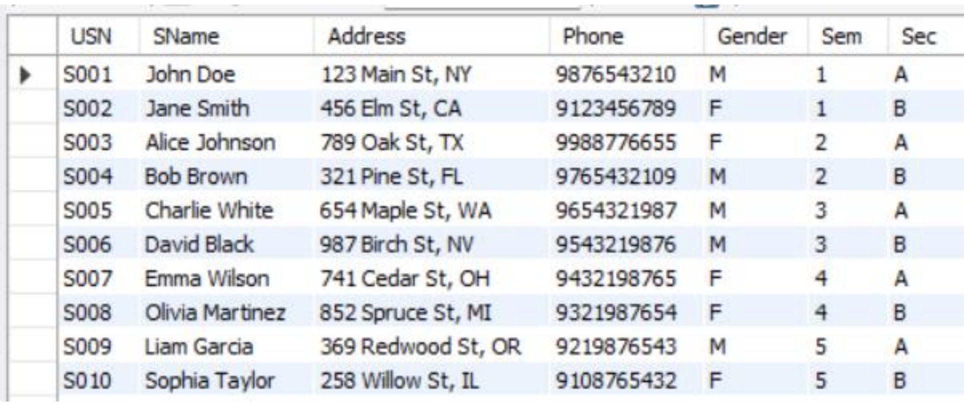


	usn	sname	gender
▶	S003	Alice Johnson	F
	S014	Isabella King	F
	S004	Bob Brown	M

5. Write SQL queries to:

1. Retrieve Students and Their Semester Details using NATURAL JOIN.

```
SELECT S.USN, S.SName, S.Address, S.Phone, S.Gender, SS.Sem, SS.Sec
FROM STUDENT S
NATURAL JOIN CLASS
NATURAL JOIN SEMSEC SS;
```

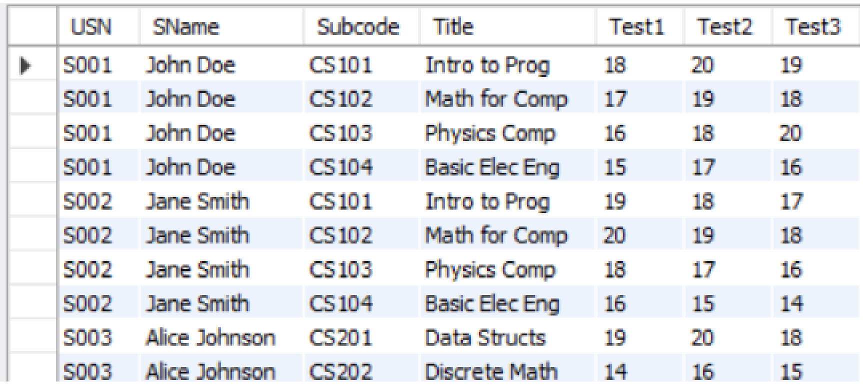


	USN	SName	Address	Phone	Gender	Sem	Sec
▶	S001	John Doe	123 Main St, NY	9876543210	M	1	A
	S002	Jane Smith	456 Elm St, CA	9123456789	F	1	B
	S003	Alice Johnson	789 Oak St, TX	9988776655	F	2	A
	S004	Bob Brown	321 Pine St, FL	9765432109	M	2	B
	S005	Charlie White	654 Maple St, WA	9654321987	M	3	A
	S006	David Black	987 Birch St, NV	9543219876	M	3	B
	S007	Emma Wilson	741 Cedar St, OH	9432198765	F	4	A
	S008	Olivia Martinez	852 Spruce St, MI	9321987654	F	4	B
	S009	Liam Garcia	369 Redwood St, OR	9219876543	M	5	A
	S010	Sophia Taylor	258 Willow St, IL	9108765432	F	5	B

2. Retrieve Student Marks with Subject Details using Equijoin.
First Insert the student without Marks

INSERT INTO Student (USN, SName, Address, Phone, Gender) VALUES ('S021', 'David white', '786 Cringe St, NV', '9786543416', 'M');

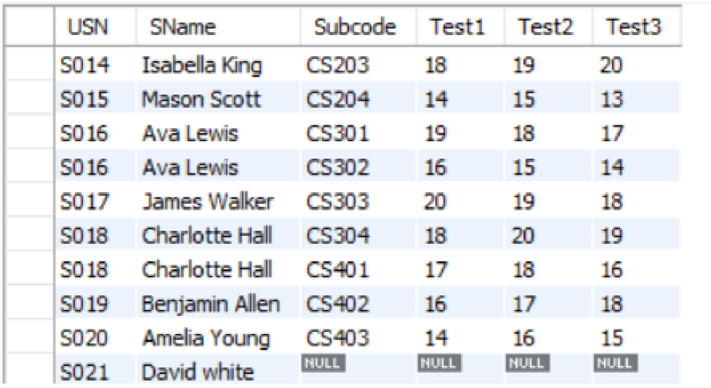
SELECT S.USN, S.SName, IA.Subcode, SUB.Title, IA.Test1, IA.Test2, IA.Test3
FROM STUDENT S
JOIN IAMARKS IA ON S.USN = IA.USN
JOIN SUB ON IA.Subcode = SUB.Subcode;



	USN	SName	Subcode	Title	Test1	Test2	Test3
▶	S001	John Doe	CS101	Intro to Prog	18	20	19
	S001	John Doe	CS102	Math for Comp	17	19	18
	S001	John Doe	CS103	Physics Comp	16	18	20
	S001	John Doe	CS104	Basic Elec Eng	15	17	16
	S002	Jane Smith	CS101	Intro to Prog	19	18	17
	S002	Jane Smith	CS102	Math for Comp	20	19	18
	S002	Jane Smith	CS103	Physics Comp	18	17	16
	S002	Jane Smith	CS104	Basic Elec Eng	16	15	14
	S003	Alice Johnson	CS201	Data Structs	19	20	18
	S003	Alice Johnson	CS202	Discrete Math	14	16	15

3.Retrieve All Students and Their Marks (Including Students Without Marks) using Left Outer Join.

SELECT S.USN, S.SName, IA.Subcode, IA.Test1, IA.Test2, IA.Test3
FROM STUDENT S
LEFT OUTER JOIN IAMARKS IA ON S.USN = IA.USN;

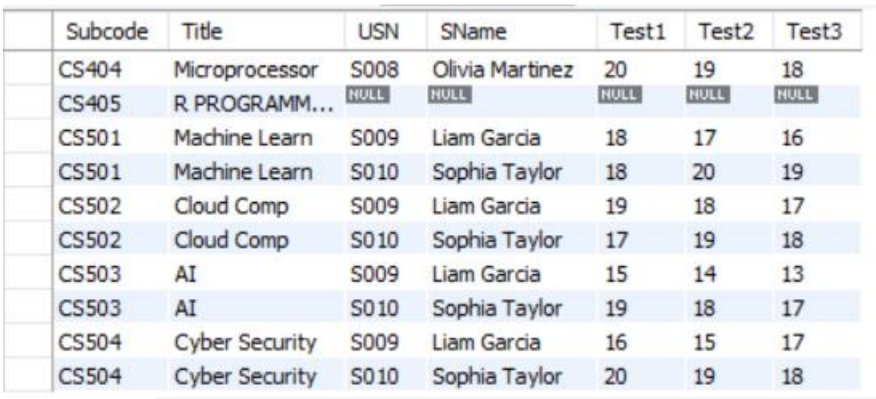


	USN	SName	Subcode	Test1	Test2	Test3
	S014	Isabella King	CS203	18	19	20
	S015	Mason Scott	CS204	14	15	13
	S016	Ava Lewis	CS301	19	18	17
	S016	Ava Lewis	CS302	16	15	14
	S017	James Walker	CS303	20	19	18
	S018	Charlotte Hall	CS304	18	20	19
	S018	Charlotte Hall	CS401	17	18	16
	S019	Benjamin Allen	CS402	16	17	18
	S020	Amelia Young	CS403	14	16	15
	S021	David white	NULL	NULL	NULL	NULL

4. Retrieve All Subjects and Their Students (Including Subjects Without Students) using Right Outer Join.

   First Insert the SUBJECT not mapped to any student

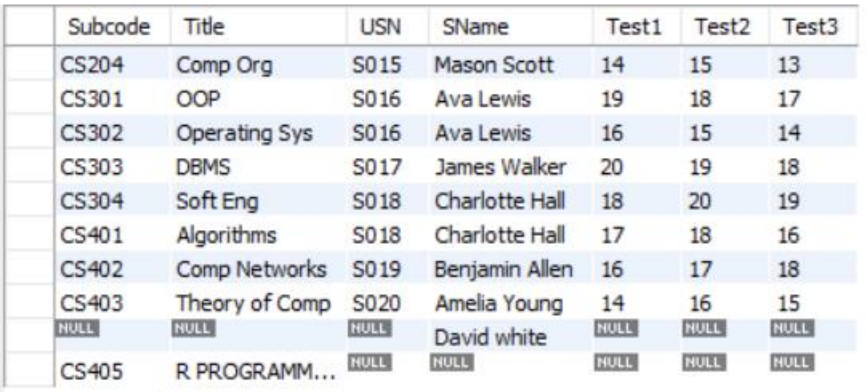
```
INSERT INTO SUB (Subcode, Title, Sem, Credits) VALUES ('CS405', 'R PROGRAMMING', 2, 3);
SELECT SUB.Subcode, SUB.Title, IA.USN, S.SName, IA.Test1, IA.Test2, IA.Test3
FROM IAMARKS IA
RIGHT OUTER JOIN SUB ON IA.Subcode = SUB.Subcode
LEFT JOIN STUDENT S ON IA.USN = S.USN;
```



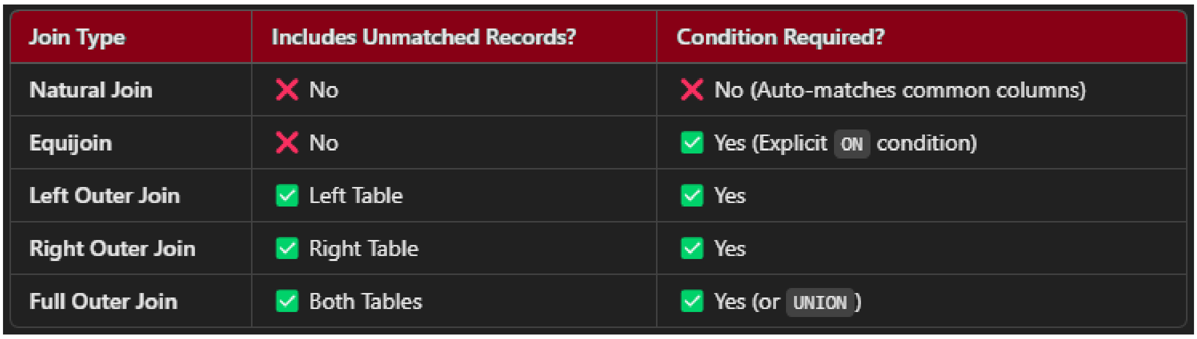
Subcode	Title	USN	SName	Test1	Test2	Test3
CS404	Microprocessor	S008	Olivia Martinez	20	19	18
CS405	R PROGRAMM...	NULL	NULL	NULL	NULL	NULL
CS501	Machine Learn	S009	Liam Garcia	18	17	16
CS501	Machine Learn	S010	Sophia Taylor	18	20	19
CS502	Cloud Comp	S009	Liam Garcia	19	18	17
CS502	Cloud Comp	S010	Sophia Taylor	17	19	18
CS503	AI	S009	Liam Garcia	15	14	13
CS503	AI	S010	Sophia Taylor	19	18	17
CS504	Cyber Security	S009	Liam Garcia	16	15	17
CS504	Cyber Security	S010	Sophia Taylor	20	19	18

5. Retrieve All Subjects and Their Students (Including Subjects Without Students) using FULL Outer Join.

```
SELECT SUB.Subcode, SUB.Title, IA.USN, S.SName, IA.Test1, IA.Test2, IA.Test3
FROM STUDENT S
LEFT JOIN IAMARKS IA ON S.USN = IA.USN
LEFT JOIN SUB ON IA.Subcode = SUB.Subcode

UNION

SELECT SUB.Subcode, SUB.Title, IA.USN, S.SName, IA.Test1, IA.Test2, IA.Test3
FROM STUDENT S
RIGHT JOIN IAMARKS IA ON S.USN = IA.USN
RIGHT JOIN SUB ON IA.Subcode = SUB.Subcode;
```



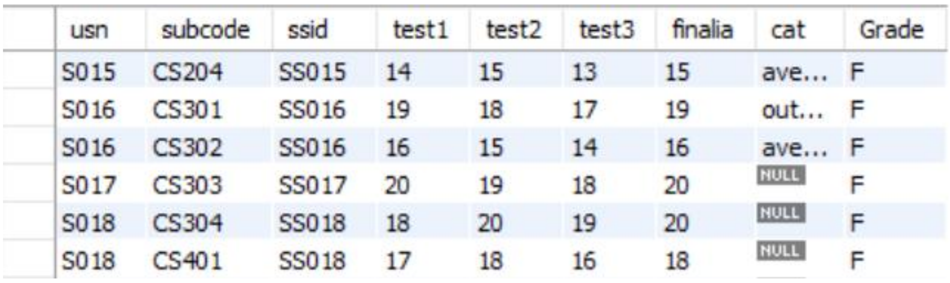
Subcode	Title	USN	SName	Test1	Test2	Test3
CS204	Comp Org	S015	Mason Scott	14	15	13
CS301	OOP	S016	Ava Lewis	19	18	17
CS302	Operating Sys	S016	Ava Lewis	16	15	14
CS303	DBMS	S017	James Walker	20	19	18
CS304	Soft Eng	S018	Charlotte Hall	18	20	19
CS401	Algorithms	S018	Charlotte Hall	17	18	16
CS402	Comp Networks	S019	Benjamin Allen	16	17	18
CS403	Theory of Comp	S020	Amelia Young	14	16	15
NULL	NULL	NULL	David white	NULL	NULL	NULL
CS405	R PROGRAMM...	NULL	NULL	NULL	NULL	NULL



Join Type	Includes Unmatched Records?	Condition Required?
Natural Join	❌ No	❌ No (Auto-matches common columns)
Equijoin	❌ No	✅ Yes (Explicit `ON` condition)
Left Outer Join	✅ Left Table	✅ Yes
Right Outer Join	✅ Right Table	✅ Yes
Full Outer Join	✅ Both Tables	✅ Yes (or `UNION`)

6. Calculate the Grade based on finalIA marks and update Grade column of the corresponding table for all students using user defined function and stored procedure.

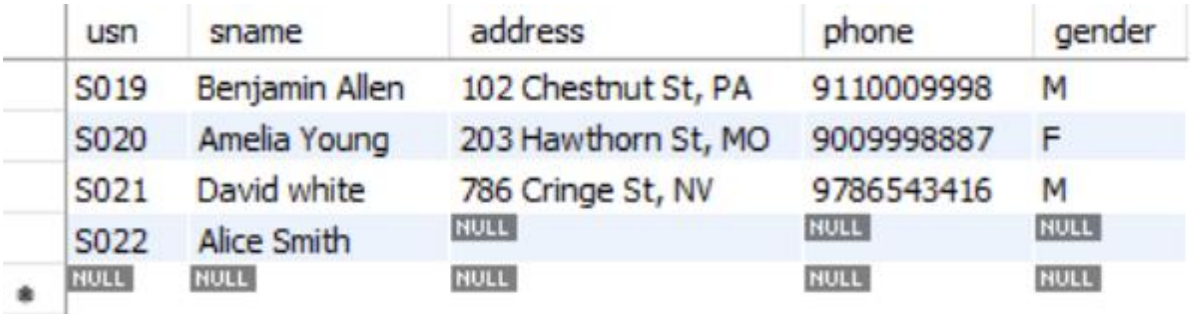
```
ALTER TABLE iamarks ADD COLUMN Grade VARCHAR(2);

DELIMITER $$

CREATE FUNCTION Calculate_Grade(FinalIA INT)
RETURNS VARCHAR(2)
DETERMINISTIC
BEGIN
   DECLARE grade VARCHAR(2);
   IF FinalIA >= 90 THEN
      SET grade = 'A+';
   ELSEIF FinalIA >= 80 THEN
      SET grade = 'A';
   ELSEIF FinalIA >= 70 THEN
      SET grade = 'B';
   ELSEIF FinalIA >= 60 THEN
      SET grade = 'C';
   ELSEIF FinalIA >= 50 THEN
      SET grade = 'D';
   ELSE
      SET grade = 'F'; -- Fail
   END IF;

   RETURN grade;
END $$

DELIMITER ;

DELIMITER $$

CREATE PROCEDURE Update_Grades()
BEGIN
   UPDATE iamarks
   SET Grade = Calculate_Grade(FinalIA);
END $$

DELIMITER ;
CALL Update_Grades();
select * from iamarks;
```



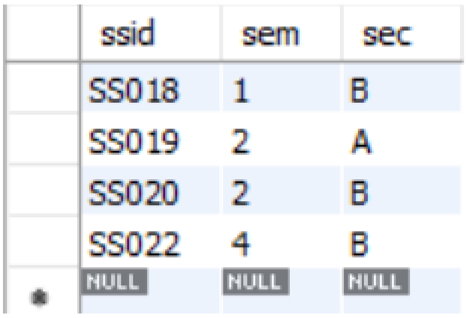
usn	subcode	ssid	test1	test2	test3	finalia	cat	Grade
S015	CS204	SS015	14	15	13	15	ave...	F
S016	CS301	SS016	19	18	17	19	out...	F
S016	CS302	SS016	16	15	14	16	ave...	F
S017	CS303	SS017	20	19	18	20	NULL	F
S018	CS304	SS018	18	20	19	20	NULL	F
S018	CS401	SS018	17	18	16	18	NULL	F

7.Write a SQL transaction to insert a new student into the STUDENT table and enroll the student in a specific class in the CLASS table. If any error occurs during the enrollment, roll back the transaction.

```
START TRANSACTION;

-- Insert a new student

INSERT INTO STUDENT (usn, sname) VALUES ('S022', 'Alice Smith');

-- Enroll the student in a semsec

INSERT INTO SEMSEC (ssid,sem,sec) VALUES ('SS022',4,'B');

-- Enroll the student in a class

INSERT INTO CLASS (usn,ssid) VALUES ('S022','SS022');

-- Commit the transaction if all inserts succeed

COMMIT;


select * from semsec;

select * from student;

select * from class;
```



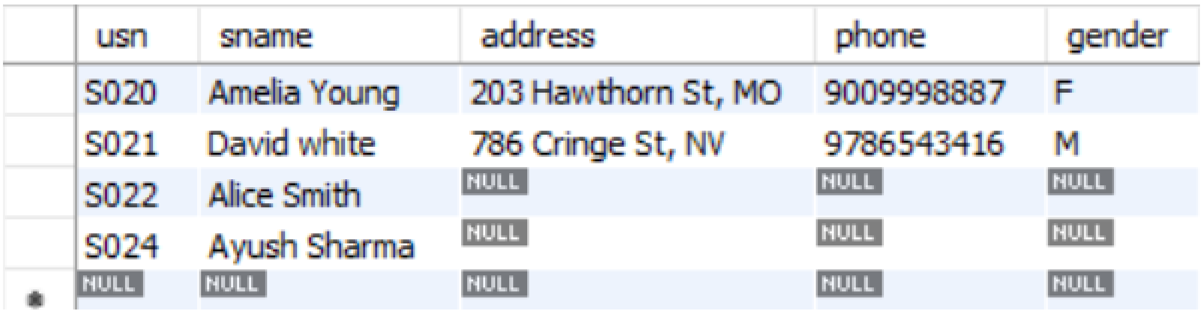
usn	sname	address	phone	gender
S019	Benjamin Allen	102 Chestnut St, PA	9110009998	M
S020	Amelia Young	203 Hawthorn St, MO	9009998887	F
S021	David white	786 Cringe St, NV	9786543416	M
S022	Alice Smith	NULL	NULL	NULL
NULL	NULL	NULL	NULL	NULL



	ssid	sem	sec
	SS018	1	B
	SS019	2	A
	SS020	2	B
	SS022	4	B
*	NULL	NULL	NULL

Create a transaction with a savepoint. If an error occurs after the savepoint, roll back to the savepoint without affecting the earlier operations (such as inserting the student record).

```
START TRANSACTION;

-- Insert a student record

INSERT INTO student (usn,sname) VALUES ('S024','Ayush Sharma');

-- Create a savepoint

SAVEPOINT sp1;

-- Attempt an operation that might fail

INSERT INTO subject (subcode, title) VALUES ('NULL', 'Mathematics'); -- Assuming
course_id cannot be NULL

-- If an error occurs, rollback to savepoint

ROLLBACK TO sp1;

-- Commit the transaction

COMMIT;

select * from student;

select * from subject;
```



	usn	sname	address	phone	gender
	S020	Amelia Young	203 Hawthorn St, MO	9009998887	F
	S021	David white	786 Cringe St, NV	9786543416	M
	S022	Alice Smith	NULL	NULL	NULL
	S024	Ayush Sharma	NULL	NULL	NULL
*	NULL	NULL	NULL	NULL	NULL

Write a DCL command to grant a user SELECT, INSERT, and UPDATE permissions on the STUDENT table.

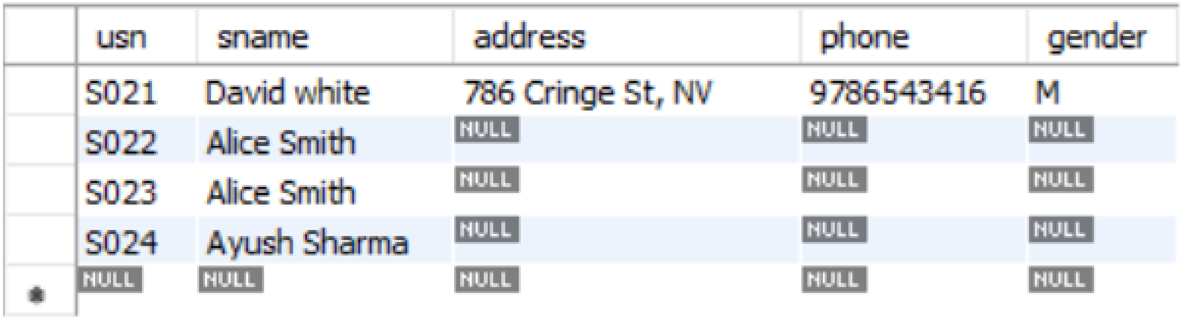
```
GRANT SELECT, INSERT, UPDATE ON STUDENT TO UserName;
```

How would you revoke all permissions granted to a specific user on the IAMARKS table?

```
REVOKE ALL PRIVILEGES ON IAMARKS FROM UserName;
```

8.Write a SQL trigger that automatically logs every insertion into the STUDENT table. The log should contain the action type (INSERT), USN, SName, and the date/time when the insertion occurred.

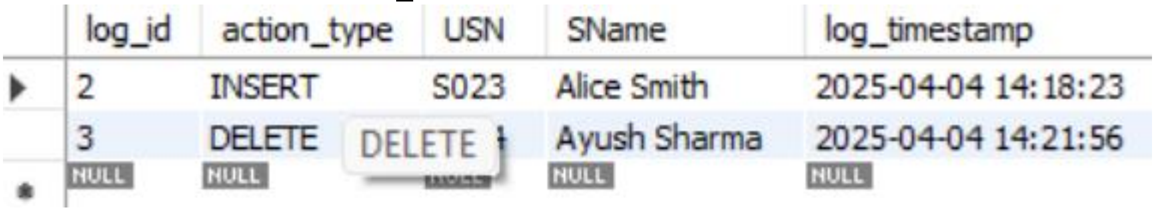
```
CREATE TABLE STUDENT_LOG (
   log_id INT AUTO_INCREMENT PRIMARY KEY,
   action_type VARCHAR(10),
   USN varchar(11),
   SName VARCHAR(15),
   log_timestamp DATETIME DEFAULT CURRENT_TIMESTAMP
);
DELIMITER $$

CREATE TRIGGER before_student_insert
BEFORE INSERT ON STUDENT
FOR EACH ROW
BEGIN
   INSERT INTO STUDENT_LOG (action_type, USN, SName, log_timestamp)
   VALUES ('INSERT', NEW.USN, NEW.SName, NOW());
END $$

DELIMITER ;
INSERT INTO STUDENT (USN, SName) VALUES ('S023', 'Alice Smith');
```



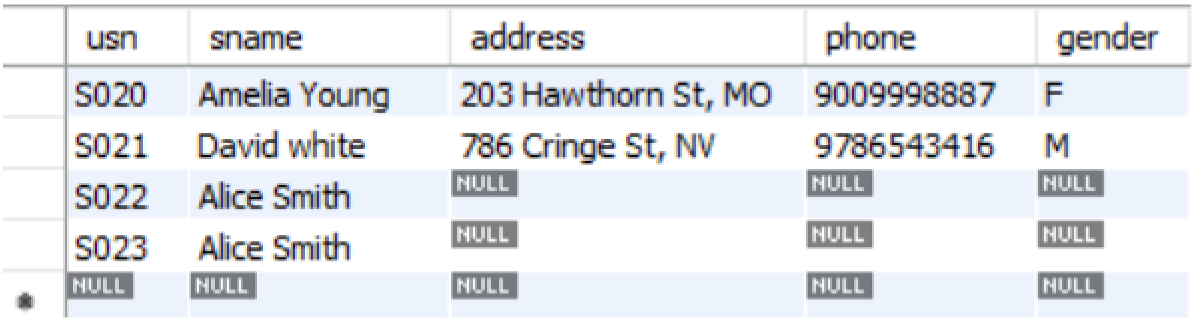
	usn	sname	address	phone	gender
	S021	David white	786 Cringe St, NV	9786543416	M
	S022	Alice Smith	NULL	NULL	NULL
	S023	Alice Smith	NULL	NULL	NULL
	S024	Ayush Sharma	NULL	NULL	NULL
*	NULL	NULL	NULL	NULL	NULL

Create a SQL trigger that logs the details of any student who is deleted from the STUDENT table. The log should include the ActionType (DELETE), USN, SName, and the date/time of the deletion.

```
CREATE TRIGGER after_student_delete

AFTER DELETE ON STUDENT

FOR EACH ROW

BEGIN

   INSERT INTO STUDENT_LOG (action_type, USN, SName, log_timestamp)

   VALUES ('DELETE', OLD.USN, OLD.SName, NOW());

END $$


DELIMITER ;


DELETE FROM STUDENT WHERE USN = 'S024';

select * from student;

SELECT * FROM STUDENT_LOG;
```



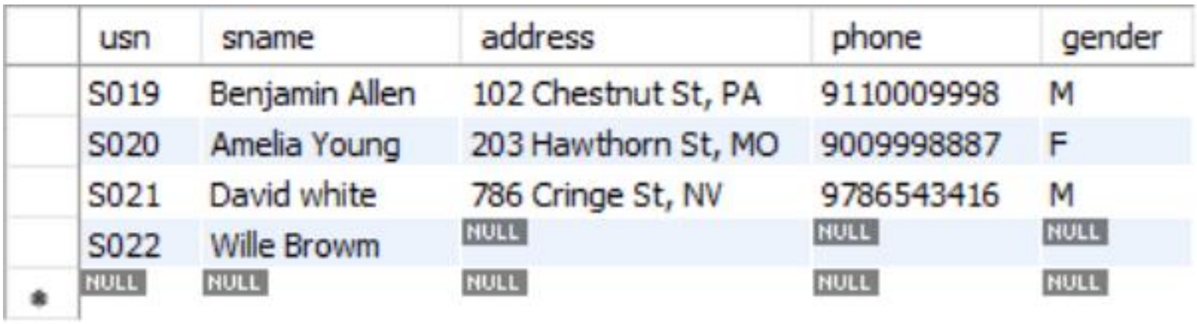
	log_id	action_type	USN	SName	log_timestamp
▶	2	INSERT	S023	Alice Smith	2025-04-04 14:18:23
	3	DELETE	[illegible]	Ayush Sharma	2025-04-04 14:21:56
*	NULL	NULL	NULL	NULL	NULL



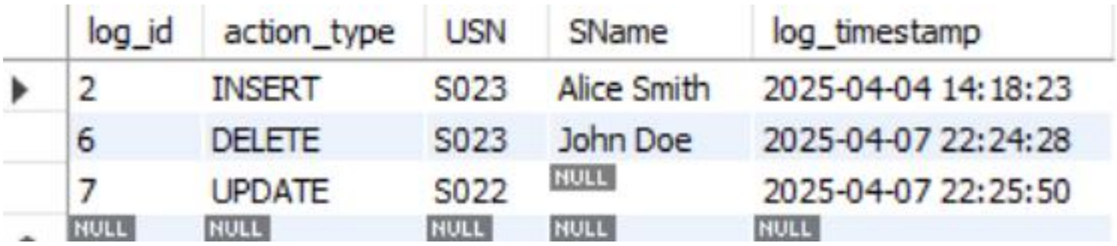
	usn	sname	address	phone	gender
	S020	Amelia Young	203 Hawthorn St, MO	9009998887	F
	S021	David white	786 Cringe St, NV	9786543416	M
	S022	Alice Smith	NULL	NULL	NULL
	S023	Alice Smith	NULL	NULL	NULL
*	NULL	NULL	NULL	NULL	NULL

Write a SQL trigger that logs every update made to the name of a student in the STUDENT table. The log should capture USN, SName, and the date/time when the update occurred.

```
DELIMITER $$


CREATE TRIGGER after_student_update
AFTER UPDATE
ON STUDENT
FOR EACH ROW
BEGIN
   INSERT INTO STUDENT_LOG (action_type, USN, SName, log_timestamp)
   VALUES ('UPDATE', OLD.USN, SName, NOW());
END $$


DELIMITER ;
UPDATE STUDENT  SET SName = 'Wille Brown'  WHERE USN = 'S022';
select * from student_log;
select * from student;
```



	usn	sname	address	phone	gender
	S019	Benjamin Allen	102 Chestnut St, PA	9110009998	M
	S020	Amelia Young	203 Hawthorn St, MO	9009998887	F
	S021	David white	786 Cringe St, NV	9786543416	M
	S022	Wille Browm	NULL	NULL	NULL
*	NULL	NULL	NULL	NULL	NULL

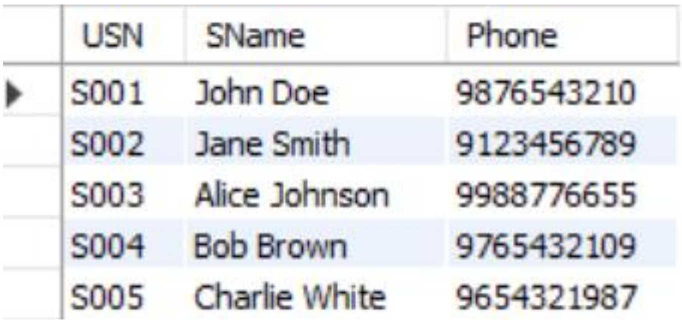


	log_id	action_type	USN	SName	log_timestamp
▶	2	INSERT	S023	Alice Smith	2025-04-04 14:18:23
	6	DELETE	S023	John Doe	2025-04-07 22:24:28
	7	UPDATE	S022	NULL	2025-04-07 22:25:50
	NULL	NULL	NULL	NULL	NULL

9.Create a view to display basic student information including USN, SName, and Phone from the STUDENT table.

CREATE VIEW student_basic_info AS SELECT USN, SName, Phone FROM STUDENT;

SELECT * FROM student_basic_info;

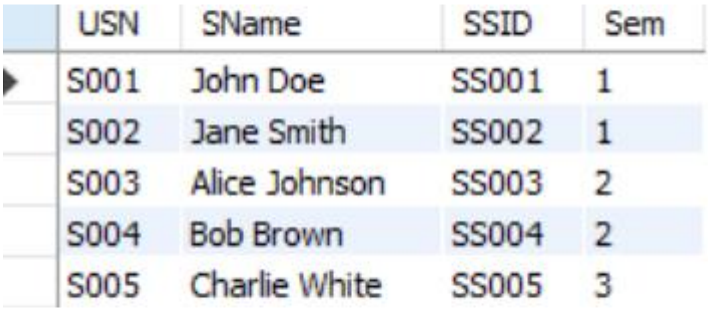


USN	SName	Phone
S001	John Doe	9876543210
S002	Jane Smith	9123456789
S003	Alice Johnson	9988776655
S004	Bob Brown	9765432109
S005	Charlie White	9654321987

Create a view to show students along with their enrolled classes, including USN, SName, SSID (class ID), and Sem (semester). This will involve a join between the STUDENT, CLASS, and SEMSEC tables.

CREATE VIEW student_class_info AS SELECT s.USN, s.SName, c.SSID, ss.Sem FROM STUDENT s JOIN CLASS c ON s.USN = c.USN JOIN SEMSEC ss ON c.SSID = ss.SSID;

SELECT * FROM student_class_info;

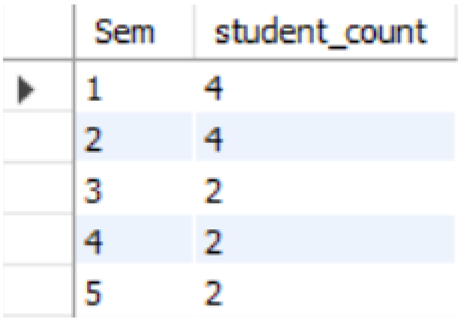


USN	SName	SSID	Sem
S001	John Doe	SS001	1
S002	Jane Smith	SS002	1
S003	Alice Johnson	SS003	2
S004	Bob Brown	SS004	2
S005	Charlie White	SS005	3

Create a view to show the total number of students in each semester, grouping by Sem from the SEMSEC table. This will involve the CLASS table to count the number of students per semester.

CREATE VIEW semester_student_count AS SELECT ss.Sem, COUNT(c.USN) AS student_count FROM SEMSEC ss JOIN CLASS c ON ss.SSID = c.SSID GROUP BY ss.Sem;

SELECT * FROM semester_student_count;

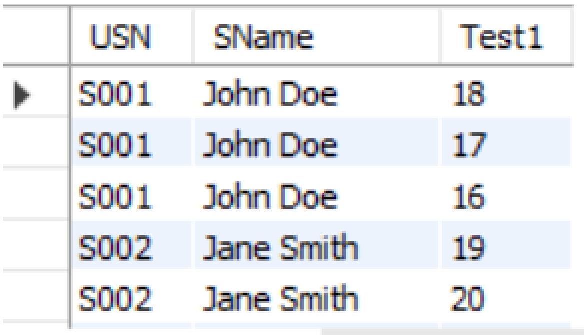


Sem	student_count
1	4
2	4
3	2
4	2
5	2

Create a view to show only the students who have scored above 15 in Test1 in the IAMARKS table. Include the USN, SName, and Test1 score.

CREATE VIEW high_scorers AS SELECT s.USN, s.SName, i.Test1 FROM STUDENT s JOIN IAMARKS i ON s.USN = i.USN WHERE i.Test1 > 15;

SELECT * FROM high_scorers;



USN	SName	Test1
S001	John Doe	18
S001	John Doe	17
S001	John Doe	16
S002	Jane Smith	19
S002	Jane Smith	20