1. Add gender column for the student table. It holds two value (male or female).

ALTER TABLE student

ADD COLUMN gender ENUM('Male', 'Female');

1. Add birth date column for the student table.

ALTER TABLE student

ADD COLUMN birthdate DATE;

1. Delete the name column and replace it with two columns first name and last name.

ALTER TABLE student

DROP COLUMN name,

ADD COLUMN fname VARCHAR(100),

ADD COLUMN lname VARCHAR(100);

1. Add foreign key constrains in Your Tables with options on delete cascaded .

alter table student\_subject

add foreign key (id) REFERENCES student(id)

on delete cascade ;

alter table student\_subject

add foreign key (SubjectID) REFERENCES subject(SubjectID)

on delete cascade ;

1. Update your information by changing data for (gender, birthdate, first name, last name).

UPDATE student

SET gender = 'female', birthdate = '2020-01-01', fname = 'aya', lname = 'ezzat'

where id =2;

1. Insert new student and his score in exam in different subjects as transaction.

start transaction;

insert into student (id,email,address,birthdate,fname,lname,gender)

values (8,"a@gmail.com",'mahalla','2076-05-07','ahmed','mohamed','male');

INSERT INTO student\_subject (id , SubjectID, max\_score ,date\_exam ) VALUES

('7', '2', '20','2029-01-10');

Commit;

7. Display all students’ information.

select \* from student;

8. Display male students only.

SELECT \* FROM student WHERE gender = 'Male';

9. Display the number of female students.

SELECT COUNT(\*) FROM student WHERE gender = 'Female';

10. Display male students who are born before 1991-10-01.

SELECT \* FROM student WHERE gender = 'Male' AND birthdate < '1991-10-01';

11. Display subjects and their max score sorted by max score.

SELECT description, MAX(MaxScore) AS maxScore

FROM subject

GROUP BY description

ORDER BY MaxScore DESC;

12. Display the subject with highest max score

SELECT description, MAX(MaxScore) AS maxScore

FROM subject

GROUP BY description

ORDER BY MaxScore DESC

Limit 1;

13. Display students’ names that begin with A.

SELECT \* FROM student WHERE fname LIKE 'A%';

14. Display the number of students’ their name is “Mohammed”

SELECT COUNT(\*) FROM student WHERE fname = 'Mohammed';

15. Display the number of males and females.

SELECT gender, COUNT(\*) FROM student GROUP BY gender;

16. Display the repeated first names and their counts if higher than 2.

SELECT fname, COUNT(\*) AS ncount

FROM student

GROUP BY fname

HAVING COUNT(\*) > 2;

17. Create a view for student names with their Tracks names which is belong to it.

At first we will create a new table called “track”

CREATE TABLE track (

trackId INT AUTO\_INCREMENT PRIMARY KEY,

trackName VARCHAR(255)

);

ALTER TABLE student

ADD COLUMN trackId INT;

alter table subject

add column trackId int;

ALTER TABLE student

ADD CONSTRAINT fk\_stutrack

FOREIGN KEY (trackId) REFERENCES track(trackId)

ON DELETE CASCADE;

ALTER TABLE subject

ADD CONSTRAINT fk\_subtrack

FOREIGN KEY (trackId) REFERENCES track(trackId)

ON DELETE CASCADE;

CREATE VIEW stutrack\_view AS

SELECT s.id, s.fname, s.lname, t.trackName

FROM student s

JOIN track t ON s.trackId = t.trackId;

18. Create a view for Tracks names and the subjects which is belong/study to it.

CREATE VIEW tracksub\_view AS

SELECT t.trackName, s.description

FROM track t

JOIN subject s ON t.trackId = s.trackId;

19. Create a view for student names with their subject's names which will study.

CREATE VIEW student\_subject\_view AS

SELECT s.id AS student\_id, s.fname AS first\_name, s.lname AS last\_name, sub.description

FROM student s

JOIN student\_subject ss ON s.id = ss.id

JOIN subject sub ON ss.id = sub.SubjectID;

20. Create a view for students’ names, their score and subject name.

CREATE VIEW studentScores\_view AS

SELECT s.id AS student\_id, s.fname AS first\_name, s.lname AS last\_name, sc.MaxScore, sub.description

FROM student s

JOIN MaxScore sc ON s.id = sc.student\_id

JOIN subject sub ON sc.SubjectID = sub.SubjectID;

21. Create a temporary view for all subjects with their max\_score.

CREATE TEMPORARY VIEW maxScore\_view AS

SELECT SubjectID, MAX(MaxScore) AS max\_score

FROM student\_subject

GROUP BY SubjectID;

22. Delete students their score is lower than 50 in a particular subject exam.

DELETE FROM student\_subject

WHERE SubjectID = '1' AND max\_score < 50;