

Hassan Ashas

18F-0128

CS18-D

①

Question 1 - Triggers

create or replace trigger amount_exception
before insert, update
on Media/Sponsorships

for each row

BEGIN

IF :new.sponsored-amount < 5000 THEN

Raise-Application-Error(-20000, 'Low value');

ELSIF :new.sponsored-amount > 90000 THEN

Raise-Application-Error(-20001, 'High Value');

END IF;

END;

/

Q1 B) VIEW

create view affilate-view as
select P-Affiliation from
Participants

Group by P-Name
having count(*)

> (select avg (select
count(*) from
participants
group by P-Name))

② Hassan

18F-0128

CS18-D

Question 3

③

Hassan
18F-0128
CS18-0

(1) Candidate Keys

→ Customer:-

cid, phonenumber, cccn.

→ Order:-

ordertime.

(2) Primary Keys

1) cid

2) pid

3) Surrogate key

4) sid

5) No primary key.

(3) Foreign Keys

• Order:-

cid → references Customer(cid)

pid → references pizza(pid)

• Ingredients:-

pid → references Pizza(pid)

sid → references Supplies(sid)

Q3 iv) Customer:-

(4)

Hassan
18F-0128
CS18-D

cid → NOT NULL, Unique
name → NOT NULL

Pizza:-

pid → NOT NULL
size → NOT NULL
pname → NOT NULL
price → NOT NULL

Order:-

ordertime → NOT NULL, Default now()

quantity → NOT NULL

slices → NOT NULL

Supplies:-

sname → NOT NULL

amountleft → NOT NULL

unitprice → NOT NULL

Ingredient:-

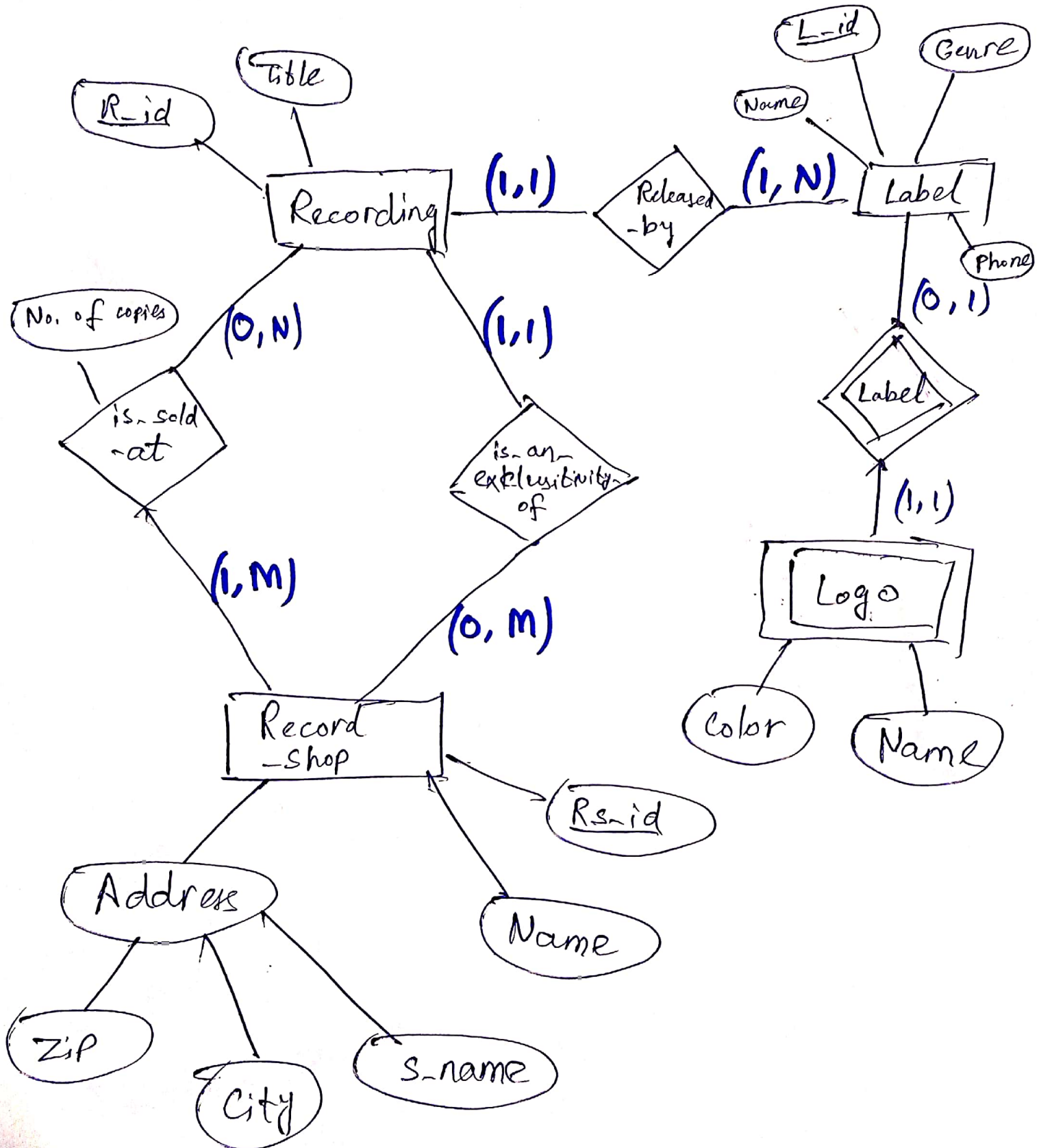
amount → NOT NULL

Question 4

ERD

5

Hassan
18F-0128
CS18-D



Question 5

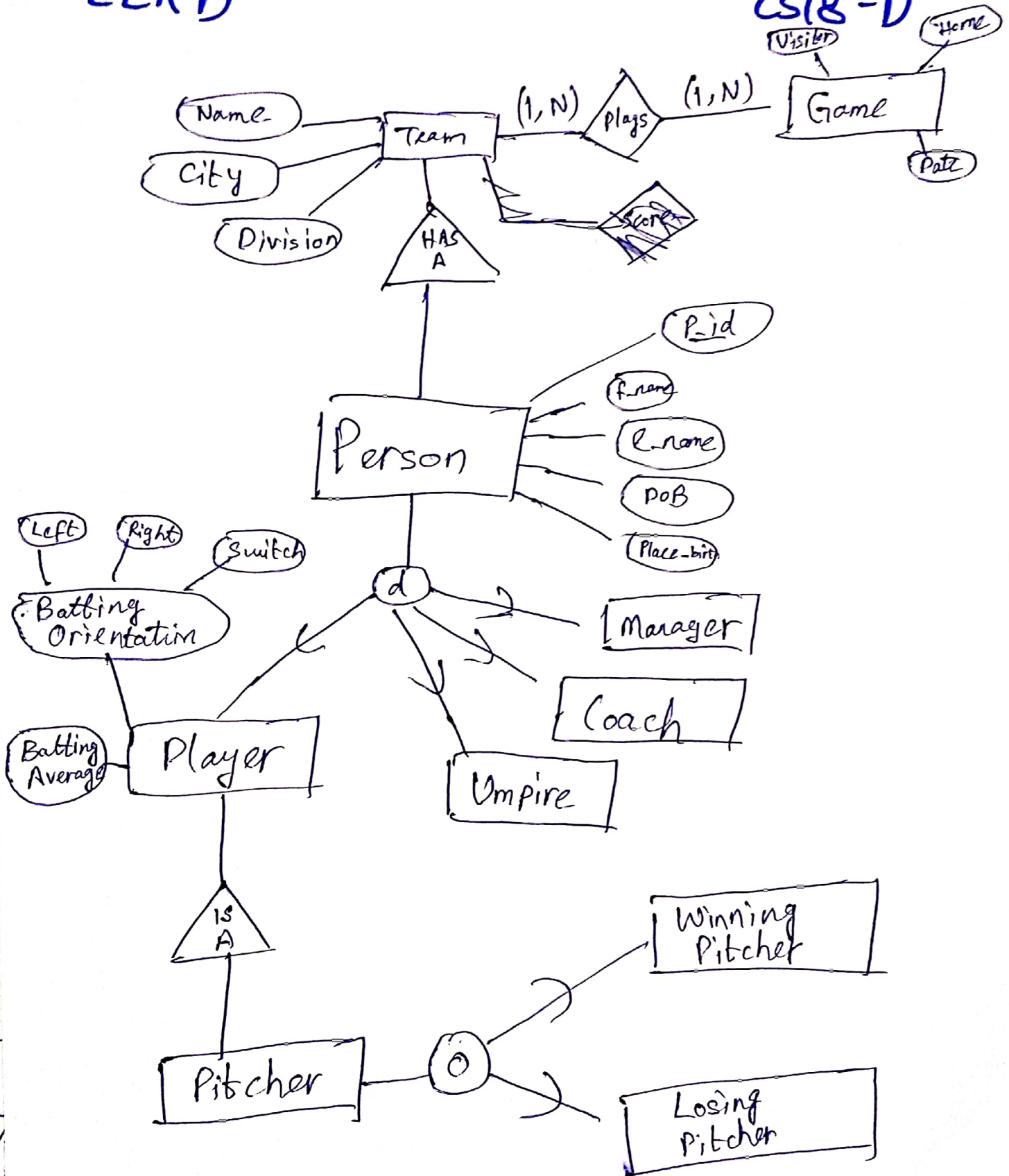
6

Hassan

18F-0128

CS18-D

EERD



Question 6 (i)

⑦

Hassan

18F-0128

CS18-D

select TOP 3

CONCAT(s.first_name + ' ' + s.last_name)

as 'Top_3_positions',

(select max(g.scores) from
graded g

where g.student_id = s.student_id

AND g.exam_id = e.exam_id)

as 'total_score'

from exam e, student s

GROUP BY total_score desc

(ii) Select s.student_id, g.exam_id, g.scores
from student s inner join graded g
on s.student_id = g.student_id
where g.scores < (select AVG(scores)
from graded h where
g.exam_id = h.exam_id)

Question 8 → Graph ^⑧ Hassan
18F-0128
CS18-D

