MCQ - choose all correct answers.

1. ANS: A, C, D
A → True there is no participation constraint in its relationship to Disease.
B → False, this will not let us keep track of a drug belonging to multiple classes.
C → True (it is not in the data requirements)
D → True, there is nothing in the Drug entity set that keeps track of the previous value of status
E → False drug has many-many relationship with Disease

2. ANS: B, C
A → False, No such restrictions on weak entity sets
B → True, many-many relationship
C → True, dosage is an attribute of the Trial entity set.
D → False, information is derivable by counting the number of negative effects in the relationship between Patients and Trial.
E → False, participation constraint means at the least one, not more than one.

3. ANS: A, D
A → True, participation constraints without key constraints cannot be enforced in relational model.
B → False, that will not keep track of effects from different trials/drugs
C → False, each Trial (Phase 0, 1.. etc) can have its own dosage
D → True, there are indeed no constraints defined.
E → False, Foreign keys are always connected to the Relation of the other entity in the relationship (in this case Trial). Otherwise you can have anomalous values.

4. ANS: D, E
A → False, we need druguses and drug for this.
B → False, a drug may not be any class in which case we cannot use drugclasses.
C → False, drugclasses and druguses are sufficient.
D → True, (Eg., 2 drugs, both belonging to one class)
E → True, (Eg. 1 drug that belongs to two classes)

5. ANS: C, E
A → False, you can also create it before drugclasses and druguses.
B → False, that is not an issue.
C → True
D → False, there is only one candidate key, and that is (pid, drugid, phase).
E → True, because later has a FK referencing the PK of the former.

6. ANS: A, C
A → True,
B → False (projection removes columns before join is done)

C → True, joining with diseases is redundant, but do not change the output.
D → False, like, no way, incorrect use of union …
E → False (projection removes columns before join is done)

7. ANS: B, D
A → False, SQL results in a cartesian join, not going to produce correct results
B → True
C → False, SQL results in a cartesian join, not going to produce correct results
D → True (corelated subquery version of B)
E → False, SQL results in a cartesian join, not going to produce correct results

8. ANS: A
Because of math.
Size of a tuple = 2*8 + 7 + 8 = 31
Number of tuples = 8 drugs * 5 trials * 150 patients = 6000
Total bytes = 6,000 * 31 = 186,000
Pages required = 186,000 / (4000 * 0.75) = 62

Open Questions

9. ANS:

proj pname, addr (patients natjoin proj pid (Sel drugid = 1223 and phase = ‘Phase 0’ (patienttrial)) INTERSECT proj pid (Sel drugid = 1223 and phase = ‘Phase 4’ (patienttrial)))

```
SELECT pname, addr from patient where pid IN
(
    SELECT pid FROM patienttrial WHERE drugid = 1223 and phase = ‘Phase 0’
        INTERSECT
    SELECT pid FROM patienttrial WHERE drugid = 1223 and phase = ‘Phase 4’
)
```

10. ANS:

```
SELECT drugname
FROM drugs
WHERE drugid IN
(
    SELECT drugid
    FROM patienttrial
    WHERE effect = 'negative'
    AND pid IN
    (
        SELECT pid
        FROM patient
        WHERE pname = 'John Doe'
    )
)
```

11. ANS:

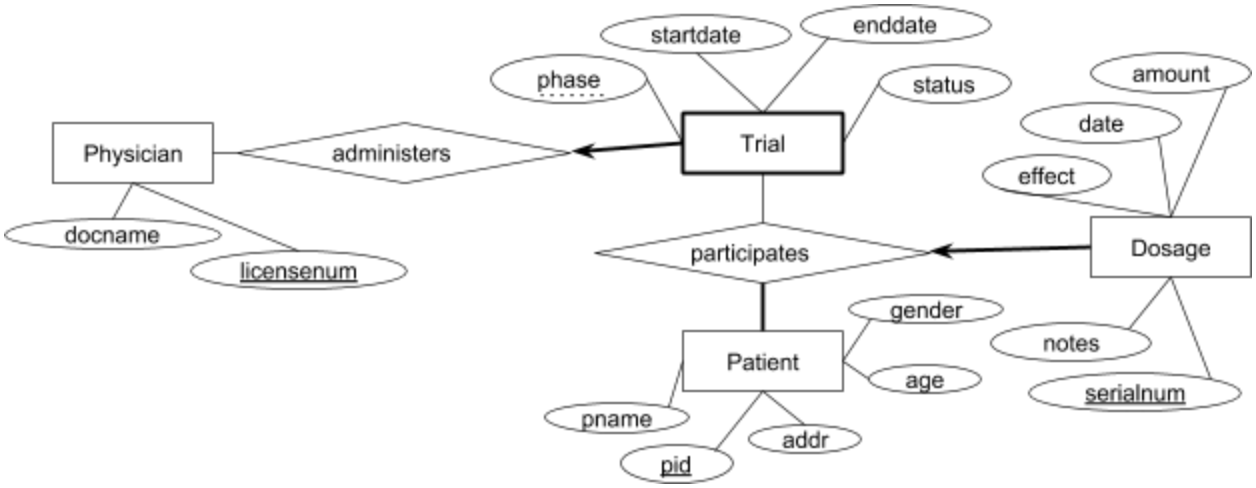
```
SELECT drugname
FROM drug d INNER JOIN druguses u
  ON d.drugid = u.drugid
WHERE d.status = 'production' AND u.diseasename = 'multiple sclerosis'
        AND drugid in
        (
                SELECT drugid
                FROM patienttrial
                WHERE effect = 'positive'
                GROUP BY drugid
                HAVING COUNT(*) >= 100
                -- If you used distinct patient ids …
                --HAVING COUNT(DISTINCT pid) >= 100
                -- additional logic if you may chosen to avoid all patients
                -- with a negative effect ever.

                EXCEPT

                SELECT drugid
                FROM patienttrial
                WHERE effect = 'negative'
                GROUP BY drugid
                HAVING COUNT(*) >= 10
                -- If you used distinct patient ids …
                --HAVING COUNT(DISTINCT pid) >= 10
        )
```

If you used < 10 with AND / INTERSECT in your logic, it will not work in cases where you have a drug with no negative effects.

12 ANS:
patient(pid, pname, addr, age, gender)

(a) Avg record length = 8 + 15 + 30 + 2 + 1 = 56 bytes
(b) Avg number of records in a data page = 4000 bytes page size * 0.75 fill / 56 byte record size = 54 records
(c) number of data entries = (50 age * 3 gender) = 150 data entries.
6000 patients / 150 data entries = 40 rids / data entry
(d) Size of a data entry = (age=2 bytes+gender=1byte) + (5 bytes rid * 40 rids / data entry) = 203 bytes.
data entries in a page = 4000 bytes page size * 0.75 fill / 203 bytes per data entry = 15
number of leaf nodes = number of data entries / data entries in a page = 150/15 = 10
(e) Number of data pages of patients table = 6000 patients * 56 bytes per record / (4000 bytes page size * 0.75 fill) = 112 data pages. Since we have 40 rids per data entry, we may potentially have to read 40 data pages for a given (age, gender) combination. Therefore using the index is advantageous as long as the total number of rids to be read is less than 112. Thus the number of (age, gender) combination qualifying should be less than 112 data pages / 40 rids per data entry, i.e., less than 2.8. So basically we can have only a maximum of 2 (age, gender) combination qualifying as part of the query, which is possible when X is 49.

13. ANS:



physician(<u>licensenum</u>, docname)
trial(<u>drugid</u>, <u>phase</u>, startdate, enddate, status, licensenum)
drugid references drug, licensenum references physician
dosage(pid, drugid, phase, <u>serialnum</u>, date, amount, effect, notes)
(drugid,phase) references trial, pid references patient,

patitenttrial relation is replaced by dosage.