Databases C130 2017/18

========= Question 1 =========

--- 1a ---

i)

Retrieve name of all renters who have made at least one rental

SELECT r.Name

FROM (

SELECT r.Name, count(MovieID) as count

FROM RENTERS r LEFT JOIN DVD\_RENTAL d

GROUP BY r.Name

)

WHERE count > 0

ORDER BY r.Name ASC

ii)

Subuery (StoreName, count(Non-Dreamworks), count(Dreamworks)):

SELECT StoreName

FROM (

(SELECT StoreName, count(\*) as NonDW\_count

FROM DVD\_STORE JOIN DVD\_RENTAL JOIN MOVIE

WHERE MOVIE.ProducerName <> "DreamWorks"

GROUP BY StoreName)

JOIN

(SELECT StoreName, count(\*) as DW\_count

FROM DVD\_STORE JOIN DVD\_RENTAL JOIN MOVIE

WHERE MOVIE.ProducerName <> "DreamWorks"

GROUP BY StoreName)

)

WHERE DW\_count > 0 AND NonDW\_count = 0

ORDER BY StoreName ASC

iii)

SELECT StoreName

FROM (DVD\_RENTAL JOIN MOVIE) d1 EXCEPT (

SELECT StoreName

FROM (DVD\_RENTAL JOIN MOVIE) as d\_with\_Dreamworks

WHERE ProducerName = "DreamWorks" )

WHERE d.StoreID = d\_with\_Dreamworks.StoreID

ORDER BY StoreName ASC

iv)

Subquuery: (StoreName, count(MemberNo))

hence:

SELECT DISTINCT StoreName

FROM (SELECT StoreID, count(DISTINCT MemberNo) as NumberOfMembers

FROM DVD\_STORE JOIN DVD\_RENTAL

GROUP BY StoreID) JOIN DVD\_STORE

WHERE NumberOfMembers > 500

ORDER BY StoreName ASC

v)

SELECT MemberNo as "Member ID"

FROM (SELECT MemberNo, count(DISTINCT StoreID as StoreCount)

FROM DVD\_RENTAL

GROUP BY MemberNo)

WHERE StoreCount = 1

ORDER BY MemberNo ASC

--- 1b ---

Doctor(PK:doc\_id, Fname, Lname)

Patient(PK:pat\_id, Fname, Lname)

Hospitalize(PK:doc\_id, PK:pat\_id, AdmissionDate)

doc\_id references Doctor.doc\_id

pat\_id references Patient.pat\_id on delete cascade

MedicalExamination(PK:unique\_num, Name)

Participate(PK:pat\_id, PK:unique\_num)

pat\_id references Patient.pat\_id on delete cascade

unique\_num references MedicalExamination.unique\_num on delete cascade

DiagnosticTest(PK:Name, PK:doc\_id, PK:unique\_num)

doc\_id references Doctor.doc\_id on delete cascade

unique\_num references MedicalExamination.unique\_num on delete cascade

Decides(PK:doc\_id, unique\_num)

doc\_id references Doctor.doc\_id on delete cascade

unique\_num references MedicalExamination.unique\_num on delete cascade

Conducts(PK:doc\_id, PK:test\_name)

doc\_id references Doctor.doc\_id on delete cascade

test\_name references DiagnosticTest.Name on delete cascade

ConsistsOf(PK:exam\_no, PK:test\_name)

exam\_no references MedicalExamination.unique\_num on delete cascade

test\_name references DiagnosticTest.Name on delete cascade

========= Question 2 =========

--- 2a ---

BD->AC:

is B extraneous? yes D->C C->AB

is D extraneous? yes B->CE B->A

.. remove FD1

B->CE:

is C extraneous? no

is E extraneous? yes C->E

.. FD becomes B->C

AB->E:

is A extraneous? yes B->C C->E

.. FD becomes B->E

ABD->E:

.. B->E so remove FD

C->AB:

is A extraneous? yes B->A

.. FD becomes C->B

We have:

C->E

B->C

D->C

B->A

B->E

C->B

C->BE // E is extraneous so

Canonical Cover is:

C->B

B->ABE

D->C

--- 2b ---

E->BD

D->E

--- 2c ---

First, canonical cover:

BD->A

D->E

B->CE

Key: {BD}

3NF:

R(ABCDE)

R1(ABD)

R2(DE)

R3(BCE)

BD key is in R1

--- 2d ---

Key: {ACD} or {ADF}

R(ABCDEF)

R1(DBE) D->BE in BCNF

R2(ACDF)

R21(CF) C->F in BCNF

R22(ACD)

--- 2e ---

Key: {AD}

R(ABCDEF)

R1(BCEF) BF->CE

R2(ABDF)

R21(BDF) DF->B

R22(ADF) AD->F