Dersin Kodu: BIMU3064

Dersin Adı: Veri Tabanı Yönetim Sistemleri

Öğrenci Numaram: 1306170114

Adım: Enver

Soyadım: Usta

Ödev Numarası: Ödev2

Hangi Veritabanı: PostgreSQL

Soru-1

1. (10puan-IN/EXISTS/=SOME/ALL) ‘Ali KURT’ adlı doktorun çalıştığı hastanenin kaydını (hospital tablosundaki tüm sütunları) döndüren sorguyu IN, EXISTS ve SOME/ALL ile 3 farklı şekilde veriniz.

1.1 IN Sorgusu:

SELECT \* FROM hastane

WHERE hid in(

SELECT hid FROM doctor

WHERE doctor.adi='Ali KURT'

);

1.2 EXISTS Sorgusu:

SELECT \* FROM hastane

WHERE EXISTS(

SELECT hid FROM doctor

WHERE doctor.adi='Ali KURT' and hastane.hid=doctor.hid

);

1.3 SOME/ALL Sorgusu:

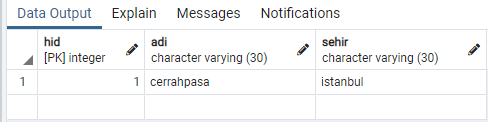
SELECT \* FROM hastane

WHERE hid=SOME(

SELECT hid FROM doctor

WHERE doctor.adi='Ali KURT'

);



Soru-2

2.(10puan-EXISTS) ‘Ali KURT’ adlı doktorun çalışmadığı hastanelerin kayıtlarını (hospital tablosundaki tüm sütunları) döndüren sorguyu NOT IN, NOT EXISTS ve SOME/ALL ile 3 farklı şekilde veriniz.

2.1 NOT IN Sorgusu:

SELECT \* FROM hastane

WHERE hid not in(

SELECT hid FROM doctor

where doctor.adi='Ali KURT'

);

2.2 NOT EXISTS Sorgusu:

SELECT \* FROM hastane

WHERE not EXISTS(

SELECT hid FROM doctor

WHERE doctor.adi='Ali KURT' and doctor.hid=hastane.hid

);

2.2 SOME/ALL Sorgusu:

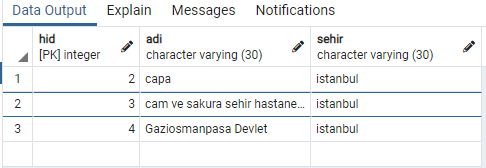
SELECT \* FROM hastane

WHERE hid!=ALL(

SELECT hid FROM doctor

WHERE doctor.adi='Ali KURT'

);



Soru-3

3.(10puan-UNION/EXCEPT/INTERSECT) a. Hem ‘Biontech’ ([testType.name](http://testtype.name/) alanını kullanınız) hem de ‘Moderna’ ([testType.name](http://testtype.name/) alanını kullanınız) testinden Negatif sonuç alan hastaların kayıtlarını INTERSECT ile listeleyiniz. b. Bir Türk (testType.origin=’tr’) veya Alman (testType.origin=’gr’) aşısından Positif sonuç alan hastaların kayırlarını UNION ile listeleyiniz. c. Hiç test yaptırmamış olan hastaların kayılarını EXCEPT/MINUS ile listeleyiniz.

3.a Sorgusu:

SELECT \* FROM hasta

WHERE pid in(

SELECT pid FROM test, testturu

WHERE testturu.adi='Biontech' and test.sonuc='neg' and test.tid=testturu.tid

)

INTERSECT

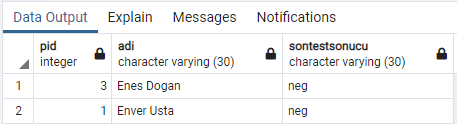
SELECT \* FROM hasta

WHERE pid in(

SELECT pid FROM test, testturu

WHERE testturu.adi='Moderna' and test.sonuc='neg' and test.tid=testturu.tid

);



3.b Sorgusu:

SELECT \* FROM hasta

WHERE pid in(

SELECT pid FROM test, testturu

WHERE test.tid=testturu.tid and testturu.orijin='gr' and sonuc='pos'

)

UNION

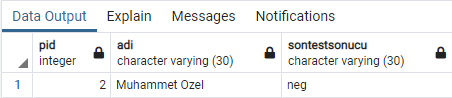
SELECT \* FROM hasta

WHERE pid in(

SELECT pid FROM test, testturu

WHERE test.tid=testturu.tid and testturu.orijin='tr' and sonuc='pos'

)



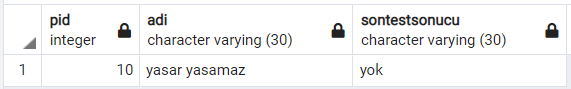
3.c Sorgusu:

SELECT \* FROM hasta

EXCEPT

SELECT hasta.\* FROM hasta, test

WHERE hasta.pid=test.pid;



Soru-4

4.(10puan-GROUP BY-HAVING) Testleri orijin (testType.orgin) ülkelerine göre gruplayarak; sadece doğruluk oranı (testType.accuracy > 80) yüzde 80’in üzerinde olan testler verileri/kayıtları kullanılarak ve ‘tr’ haricindeki ülkeler için olmak üzere; her ülke için kaç test yapıldığı (test tablosundan bulunur), testler için toplam kaç para ödendiği (testType.price), yapılan testlerin yüzde kaçının positif, yüzde kaçının negatif geldiğini listeleyiniz.

SELECT testturu.orijin, COUNT(testturu.tid), SUM(testturu.fiyat),

(100\*

(

SELECT COUNT(\*) FROM test

WHERE test.sonuc='pos' AND test.tid=testturu.tid

) / COUNT(testturu.tid)

) as pozitifperc,

(100\*

(

SELECT COUNT(\*) FROM test

WHERE test.sonuc='neg' AND test.tid=testturu.tid

) / COUNT(testturu.tid)

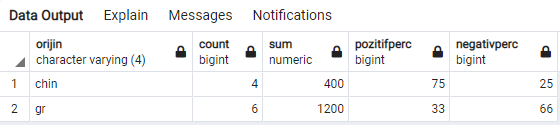
) as negativperc

FROM testturu

INNER JOIN test

ON testturu.tid=test.tid AND testturu.orijin!='tr'

GROUP BY testturu.orijin, testturu.tid;



Soru-5

5.(10puan-UNIQUE) Aynı isimli testten ([testType.name](http://testtype.name/)) iki yada daha fazla yaptırmamış (yani isim bakımından farklı testleri yaptırmış) olan hastaların kayıtlarını UNIQUE ile listeleyiniz.

SELECT \*

FROM hasta

WHERE pid NOT IN (

(SELECT pid FROM test

GROUP BY pid, tid

HAVING COUNT(tid) > 1)

UNION

(SELECT pid

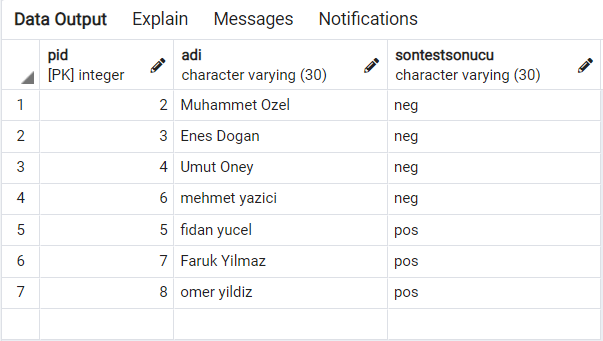
FROM hasta

WHERE pid NOT IN (

SELECT pid FROM test

))

)



Soru-6

6. (10puan WITH/FROM altsorgu) Hasta başı ortalama test sayısından daha fazla test yaptıran hastaların hid’lerini listeleyiniz. Bunu yaparken ilk iki adımı yani “hasta başı test sayısını” bir WITH alt sorgusuyla, “bunların ortalamasını” başka bir WITH sorgusuyla yazınız.

WITH hastabasitestsayisi AS(

SELECT test.pid, COUNT(test.tid) AS testsayisi FROM hasta, test

WHERE hasta.pid=test.pid

GROUP BY test.pid

),

ortalama AS(

SELECT (

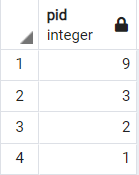
COUNT(test.pid)\*1.0

/

(SELECT COUNT(hasta.pid) FROM hasta)) AS orttestsayisi FROM test

)

SELECT pid FROM hastabasitestsayisi, ortalama WHERE testsayisi>orttestsayisi;



Soru-7

7.(10puan SELECT alt sorgusu) Hastane kayılarını listeyiniz. Fakat kayıtları listelerken her hastanedeki doktor ve yatan hasta sayılarını SELECT’in içerisine birer adet alt sorgu ekleyerek hesaplattırınız. (SELECT \*, (alt-sorgu1) doktor-sayısı, (alt-sorgu2) hasta-sayısı FROM hastane)

SELECT \*,

(

SELECT COUNT(doctor.did) FROM doctor WHERE hastane.hid=doctor.hid

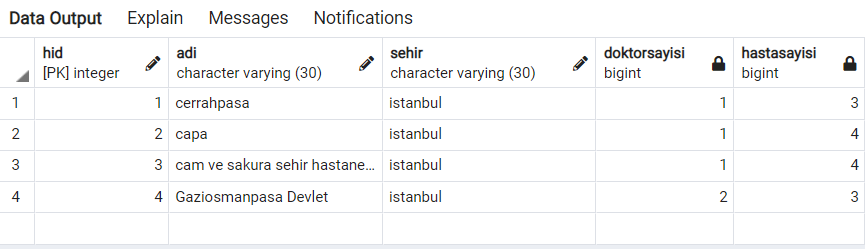
)AS doktorsayisi,

(

SELECT COUNT(hastakabul.pid) FROM hastakabul WHERE hastane.hid=hastakabul.hid

)AS hastasayisi

FROM hastane;



Soru-8

8.(10puan-HAVING altsorgusu) Uzmanlık alanı ‘dahiliye’ (specialty=’dahiliye’) olan en az bir (yani herhangi bir) doktora sahip hastaneler için hid ve o hastanede çalışan tüm doktorların sayısını listeleyiniz (HAVING içerisinde söz konusu hastanenin dahiliye uzmanı çalıştırıp7çalıştırmadığı kontrolünün yapılması gerekmektedir.)

SELECT hid, COUNT(did) AS toplamdoktor FROM doctor d1

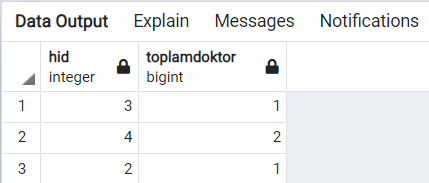
GROUP BY hid

HAVING hid=(

SELECT d2.hid FROM doctor d2

WHERE d2.uzmanlikalani='dahiliye' AND d1.hid=d2.hid

);



Soru-9

CREATE VIEW istatistik (

tarih, gunlukTestSayisi,gunlukPozitifSayisi,gunlukOlumSayisi,gunlukEntubeSayisi

) AS SELECT test.tarih, COUNT(tid),

(

SELECT COUNT(tid)

FROM test t1

WHERE sonuc='Pos'

AND t1.tarih=test.tarih

),

COUNT(olumTarih), COUNT(entubeTarih) FROM test

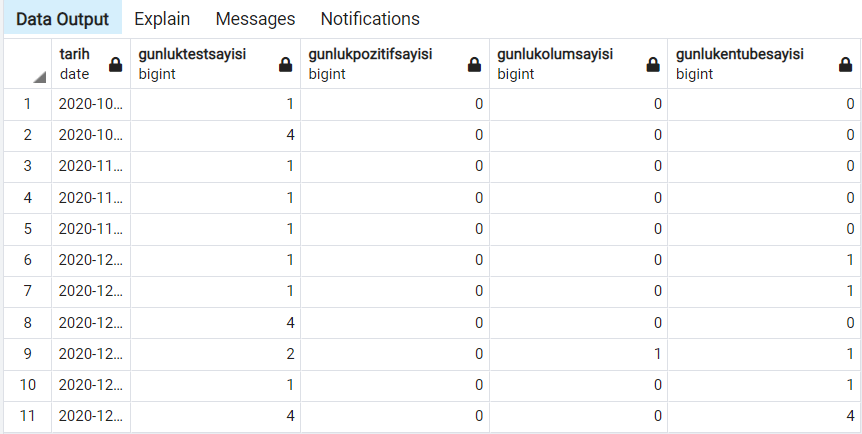
FULL OUTER JOIN hastaKabul

ON test.tarih=hastaKabul.kabulTarih

GROUP BY test.tarih

ORDER BY test.tarih;

SELECT \* FROM istatistik;



Soru-10

10. ‘Ali KURT’ adlı hastanın son testinden sonra test yaptıran hastaların kayıtlarını listeleyiniz. (Bir alt sorguda Ali KURT’un test tarihlerini döndürünüz. Hastaların test tarihini bu alt sorgudaki tarihlerle >SOME/ALL operatörünü kullanarak karşılaştırınız)

SELECT \* FROM hasta

WHERE pid IN (

SELECT pid FROM test

WHERE tarih > ALL(

SELECT MAX(tarih) FROM test

WHERE pid IN (

SELECT pid FROM hasta WHERE hasta.adi='Ali KURT'

)

GROUP BY pid

)

);

