1-)

WITH RECURSIVE rec\_prereq(course\_id, prereq\_id) AS (SELECT course\_id, prereq\_id FROM prereq WHERE course\_id = 'belirli bir dersin kodu' UNION

SELECT rec\_prereq.course\_id, prereq.prereq\_id FROM rec\_prereq, prereq WHERE rec\_prereq.prereq\_id = prereq.course\_id) SELECT \* FROM rec\_prereq;

2-)

Bu sorgu, "prereq" tablosundan başlayarak, "courseid" sütununda değeri "100" olan dersin önceden koşul derslerini ve önceden koşul derslerinin önceden koşul derslerini bulmak için recursive bir JOIN işlemi gerçekleştirir. Sonuçlar, "courseid" sütununda "100" olan dersin önceden koşul dersleri ve önceden koşul derslerinin önceden koşul derslerinden oluşan bir ağaç olarak listelenir.

3-)

Bu sorgu, öğrencilerin GPA değerlerine göre sıralandığı ve her öğrencinin kaçıncı sırada olduğunun hesaplandığı bir sorgudur.

Daha verimli bir sorgu yazmak için, RANK() fonksiyonu kullanılabilir ve ORDER BY sırasında X sütununu kullanmak yerine GPA sütununu kullanmak daha doğru sonuçlar verir ve sorgunun daha hızlı çalışmasına yardımcı olur. Önerilen sorgu şöyle olabilir:

SELECT ID, RANK() OVER (ORDER BY GPA DESC) as X

FROM student\_grades

ORDER BY GPA DESC;

4-)

Dark pastel white All

Small 20 20 24 64  
medium 23 13 9 45

Large 19 21 15 55

Total 62 54 48 328

5-)

İtem\_name color size sum

Dress dark null 20  
Dress pastel null 10

Dress white null 5

Dress null null 35

Pants dark null 20

Pants pastel null 2

Pants white null 5

Pants null null 27

Shirt dark null 14

Shirt pastel null 7

Shirt white null 28

Shirt null null 49

Skirt dark null 8

Skirt pastel null 35

Skirt white null 10

Skirt null null 53

6-)

Doküman Frekansı (DF), bir metin belgesinde belirli bir kelimenin ne sıklıkta geçtiğini ölçerek, metin arama işlemlerinde kelimenin önem derecesini belirlemeye yardımcı olur. DF ölçütü, kelimenin bir belgede ne kadar yaygın olduğuna bağlı olarak, o kelimenin arama sonuçlarında önem derecesini belirleyen etkili bir metriktir.

7-)

Raster veriler, genellikle büyük ölçekli verilerde daha yüksek performans sağlar. Özellikle, çok sayıda nokta verisi gerektiren işlemler, vektör verilerine göre daha hızlı işlenebilir.

Raster verileri, doğal kaynaklar, arazi ve hava durumu gibi fenomenlerin değişkenliğini daha iyi yansıtabilir. Bu nedenle, uzamsal veri analizi için kullanıldığında, vektör verileriyle kıyaslandığında daha doğru sonuçlar verir.

8-)

SELECT \*

FROM karayollari k, goller g, daglik\_araziler d

WHERE ST\_Intersects(k.geom, g.geom) AND ST\_Intersects(k.geom, d.geom) AND ST\_Intersects(g.geom, d.geom);

9-)

Key-value saklama sistemleri, verileri anahtar-değer çiftleri olarak saklarlar ve her bir anahtar sorgulanabilir bir indeks olarak kullanılabilir. Bu sistemlerde veri kaydetme, güncelleme ve silme işlemleri genellikle çok hızlıdır. Ayrıca, ölçeklenebilirlik açısından da avantajlıdırlar ve yatay olarak ölçeklendirilebilirler.

10-)

Google MapReduce, Apache Hadoop, Amazon Elastic MapReduce- EMR

Hive, Hcatalog, Hbase, MPP (Massively Parallel

Processing), PIG, Mahout, NoSQL ve Cassandra

11-)

Çok sayıda hafif SQL sorguları - OLTP

B-tree indekslerin yaygın kullanımı - OLTP

Veri güncellemenin olmaması - OLAP

Somut görüntülerin yaygın kullanımı - OLAP

Saklama ortamında (HDD, SSD gibi) disk sayfasının daha küçük olması - OLAP

Columnar saklama - OLAP