Analitik SQL Nedir?

Analitik SQL, veritabanı yönetim sistemlerinde kullanılan bir tür SQL sorgulama yöntemidir. Analitik SQL sorguları, veri analizi ve raporlama gibi işlemleri gerçekleştirmek için kullanılır. Bu sorgular, veri setlerindeki ilişkileri, desenleri ve trendleri analiz etmeyi sağlar.

Analitik SQL sorguları genellikle GROUP BY, HAVING, COUNT, SUM, AVG, MIN, MAX, ORDER BY gibi işlevleri kullanır. Bu sorgular, verileri gruplara ayırma, toplama, sayma, ortalama alma, minimum ve maksimum değerleri bulma gibi işlemleri gerçekleştirir. Ayrıca, analitik fonksiyonlar ve pencere fonksiyonları gibi özel SQL fonksiyonlarını da içerebilir.

***Genel Analitik SQL sorgusu:***

**SELECT**

**category,**

**COUNT(\*) AS count,**

**AVG(price) AS average\_price**

**FROM**

**products**

**GROUP BY**

**category**

**ORDER BY**

**count DESC;**

Bu sorgu, bir ürün tablosundaki ürünleri kategoriye göre gruplar ve her kategori için ürün sayısını (count) ve ortalama fiyatı (average\_price) hesaplar. Sonuçları ürün sayısına göre büyükten küçüğe sıralar.

Bu analitik SQL sorgusu, ürünlerin kategorilere göre dağılımını ve her bir kategorinin ortalama fiyatını bulmak için kullanılabilir. Böylece işletme sahipleri veya analistler, ürün portföyü hakkında bilgi edinerek, talebi yüksek olan kategorileri ve fiyat düzenlemeleri yapabilecekleri potansiyel fırsatları belirleyebilirler.

SQL Ve Analitik SQL Arasındaki Fark Nedir?

SQL, veritabanı yönetim sistemlerinde kullanılan genel bir sorgulama dili olup, veritabanları üzerinde veri ekleme, güncelleme, silme ve sorgulama gibi temel işlemleri gerçekleştirir. Analitik SQL ise, SQL dilinin analitik işlemler için kullanılan bir alt kümesidir. Analitik SQL sorguları, veriler üzerinde daha karmaşık analiz işlemlerini gerçekleştirir, veri ilişkilerini, desenlerini ve trendlerini analiz eder ve genellikle büyük veri kümeleriyle çalışırken performans optimizasyonu sağlar. Analitik SQL, gruplama, toplama, ortalama alma, minimum ve maksimum değerleri bulma gibi işlevlerin yanı sıra analitik fonksiyonlar ve pencere fonksiyonlarını da kullanır. Bu sayede daha derinlemesine analizler yapılabilir ve anlamlı içgörüler elde edilebilir.

SORGU ÖRNEKLERİ:

1. **SELECT**

**category,**

**COUNT(\*) AS count,**

**AVG(price) AS average\_price**

**FROM**

**products**

**GROUP BY**

**category;**

Bu sorgu, bir ürün tablosundaki ürünleri kategoriye göre gruplayarak, her kategori için ürün sayısını (count) ve ortalama fiyatı (average\_price) hesaplar.

1. Eğitimde bize verilen dvdrental datasetinde **SELECT \*, ROW\_NUMBER() OVER(PARTITION BY category\_id, ORDER BY film\_id) FROM film\_category**; fonksiyonu analitik bir fonksiyondur ve category\_id leri pencereleyerek o penceredeki film\_id leri sıralar.
2. DVDRENTAL.tar dosyasında içerisinde bulunan Film Tablosu bulunan "length" grup hakkında çalışma yaptım .

Agg fonksiyonların içinde bulunan 'avg' fonksiyonu kullandım. Filmlerin ortalama uzunlukları buldum. Ve grubun ismini değiştirdim.

**select avg (length) as uzunluk from film ;**

1. dvdrental.tar dosyasının içindeki örneklerde bulunan "payment" tablosu üzerinden ROUND fonksiyonu kullanım örneğinde bulundum.

**select payment\_id, ROUND(amount,0) as FİYAT\_YUVARLAMA**

**from payment;**

Bu sayede amount yani fiyat kolonunun içindeki ondalıklı sayıları virgülden sonrasını almadan yuvarlayabiliyoruz. Örneğin 0,99 olan herhangi bir ürünün fiyatı yuvarladığımızda 1 olarak gözükür. Bu bize işlem kolaylığı sağlar.