**ANKARA ÜNİVERSİTESİ MÜHENDİSLİK FAKÜLTESİ**

****

**AĞ TABANLI PARALEL DAĞITIM SİSTEMLERİ**

**Final Teslimi**

**BLM4522**

**SUDE KORKMAZ 20291275**

**Github :** [**https://github.com/CSudeKorkmaz/SQL\_Query**](https://github.com/CSudeKorkmaz/SQL_Query)

**Video linki:** [**https://drive.google.com/file/d/1rriZFiha9mTlotTIo9OUL2f9Untmzrfs/view?usp=drive\_link**](https://drive.google.com/file/d/1rriZFiha9mTlotTIo9OUL2f9Untmzrfs/view?usp=drive_link)

**Proje Özeti:**

MySQL Workbench ile Yedekleme ve Geri Yükleme İşlemleri

Bu projede, MySQL Workbench kullanılarak temel veritabanı yönetimi, yedekleme ve geri yükleme işlemleri uygulamalı olarak gerçekleştirilmiştir. İlk aşamada, sudek adında bir veritabanı oluşturulmuş ve içine kisiler adlı örnek bir tablo eklenmiştir. Bu tablo, test verileri ile doldurularak yedekleme süreci için örnek veri yapısı hazırlanmıştır.

Ardından, veritabanı MySQL Workbench’in dışa aktarma aracı kullanılarak yedeklenmiş ve daha sonra geri yükleme işlemiyle bu yedeğin doğruluğu test edilmiştir. Ayrıca, gelişmiş yedekleme senaryoları için mysqldump ve mysqlbinlog gibi komut satırı araçlarının kullanılabileceği de belirtilmiştir.

Yedekleme işlemlerinin kaydını tutmak amacıyla BackupHistory adlı bir tablo oluşturulmuş ve bu tabloya kayıt ekleyen usp\_LogBackupResult adlı bir prosedür yazılmıştır. Ek olarak, en son yedek bilgisini getiren usp\_CheckLastBackup adlı prosedür geliştirilmiştir.

Günlük yedekleme işlemlerini otomatik hale getirmek amacıyla MySQL’in EVENT yapısı kullanılarak zamanlanmış görev örneği hazırlanmış, ve bu işlemlerin takibi için event\_scheduler ayarının aktif edilmesi gerektiği vurgulanmıştır.

Sonuç olarak, bu proje kapsamında hem manuel hem de otomatik yedekleme ve geri yükleme işlemleri MySQL ortamında başarıyla uygulanmış; veri güvenliği ve sürekliliği açısından temel altyapı süreçleri test edilmiştir.

**Orijinal Sorgu:**

sql

KopyalaDüzenle

SELECT \* FROM Sales.SalesOrderHeader

WHERE YEAR(OrderDate) = 2014

**Optimizasyon Sonrası:**

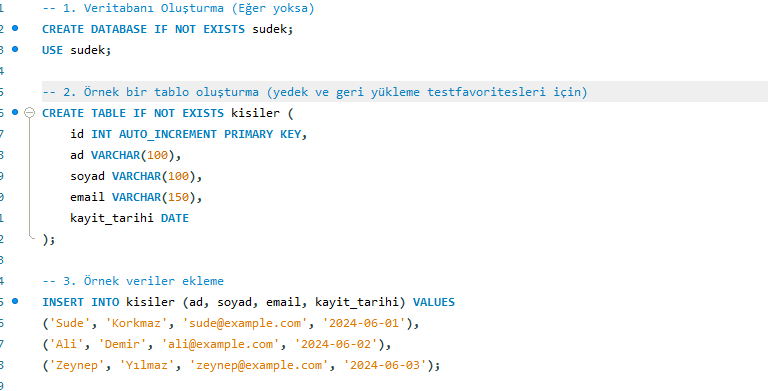
sql

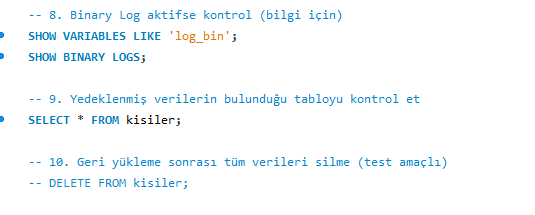
KopyalaDüzenle

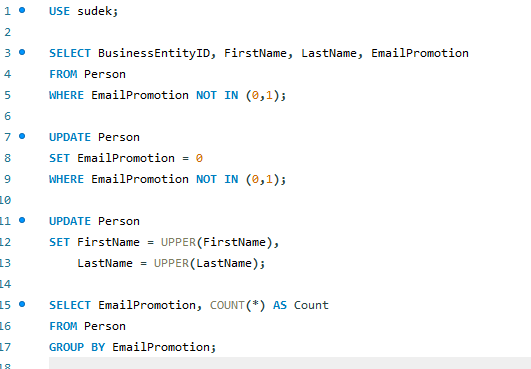
SELECT \* FROM Sales.SalesOrderHeader

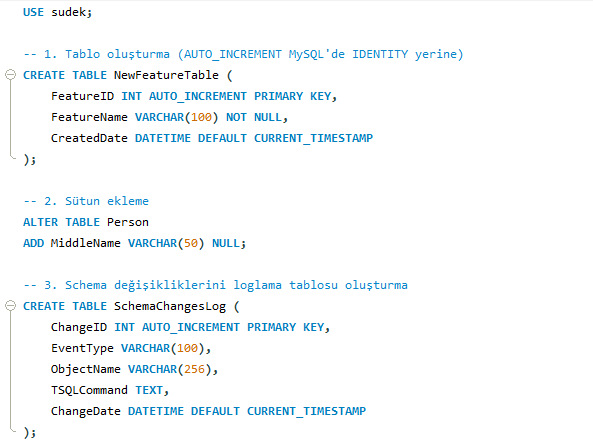
WHERE OrderDate >= '2014-01-01' AND OrderDate < '2015-01-01'

**Yer alan Kod Örnekleri:**









1. BackupHistory Tablosu Oluşturma

*İlk olarak, yedekleme işlemlerinin kayıtlarını tutmak üzere BackupHistory adlı bir tablo oluşturuyoruz. Bu tabloda, hangi veritabanının yedeği alındı, yedekleme işlemi ne zaman yapıldı, yedek dosyasının adı ne ve işlem başarılı mı gibi bilgileri depoluyoruz. Böylece yedeklerin takibini kolayca yapabiliriz. MySQL’de otomatik artan birincil anahtar, tarih için varsayılan olarak güncel zaman atanıyor.*

2. Yedekleme Sonucu Loglama Prosedürü

*Bu prosedür, her yedekleme işlemi tamamlandıktan sonra yedekleme sonucunu tabloya kaydetmek için tasarlandı. Dışarıdan veritabanı adı, yedek dosya yolu ve işlem başarılı mı bilgisi parametre olarak alınıyor. Bu sayede yedekleme durumu otomatik olarak loglanmış oluyor ve geçmişteki yedeklere kolayca erişim sağlanıyor.*

3. Son Yedek Kayıtlarını Getiren Prosedür

*Bu prosedür ise, özellikle son yedeklemeyi kontrol etmek için kullanılır. Belirli bir veritabanı için BackupHistory tablosundaki en güncel yedek kaydını döndürüyor. Böylece yedeklerin güncelliği ve başarı durumu anında kontrol edilebilir.*

4. Günlük Yedekleme Zamanlayıcısı (EVENT)

*SQL Server’daki job scheduler alternatifi olarak MySQL’de Event Scheduler kullanılır. Burada her gün saat 01:00’de tetiklenecek bir event oluşturduk. Event içinde doğrudan yedek alma komutu yoktur çünkü MySQL, yedeklemeyi SQL komutuyla değil dış araçlarla yapar. Ancak bu event ile bir prosedür çağrabilir veya dış komutları tetiklemek için ara yöntemler kullanılabilir. Örneğin, yedek alma işlemini dışarıda bir betik ile yapıp, bu prosedür veya event ile sonucu loglayabilirsin.*

5. Event Scheduler'ı Aktif Etmek

*Eventlerin aktif olarak çalışabilmesi için MySQL sunucusunda Event Scheduler özelliğini açmamız gerekir. Bu komutla scheduler aktif edilir ve oluşturduğumuz eventler planlandığı gibi tetiklenir. Sunucu yeniden başlatıldığında bu ayar kaybolabilir, kalıcı olması için MySQL konfigürasyon dosyasına da eklenmesi gerekir.*

Ek Not:

*MySQL’de veritabanı yedekleme, SQL Server’daki gibi BACKUP DATABASE komutu ile yapılmaz. Bunun yerine mysqldump gibi komut satırı araçları kullanılır. SQL kodları ile sadece yedekleme kayıtlarının tutulması, kontrolü ve yedekleme zamanlamalarının tetiklenmesi sağlanabilir. Yedek alma işlemi genellikle otomasyon betikleri ve cron job gibi sistem araçları ile yapılır.*