**MUSTAFA ESAT YÜCEL**

**221120231084**

**MOBİL PROGRAMLAMA PROJESİ**

İsteğiniz üzerine Adventure Works 2017 Veritabanı üzerinde benzersiz sorgular oluşturdum.Sorguları oluştururken özellikle gerçek hayatta da kullanılabilecek olmasına dikkat ettim çünkü bu dili biliyorum ve projeminde iş hayatımda işime yarayacak şeyleri öğrenmeme yardımcı olabilmesini istedim.Özellikle “JOIN” fonksiyonunu kullandım çünkü en çok işime yarayacak sorgu fonksiyonu bu bence.Sonda sorgularımın çalıştığı ekran görüntülerine ulaşabilirsiniz. Sorgularım ve açıklamalarım şu şekilde:

**1️. En Uzun Süre Çalışan Departman Lideri**

**HumanResources.EmployeeDepartmentHistory** tablosundaki **StartDate** ve **EndDate** sütunları kullanılarak, çalışanın bir departmanda geçirdiği toplam gün sayısı hesaplanır.

**DATEDIFF** fonksiyonu ile başlangıç ve bitiş tarihleri arasındaki fark gün bazında ölçülür.

Çalışan bilgileri, **Person.Person** ve **HumanResources.Department** tabloları ile birleştirilerek, tam isim ve departman bilgileri eklenir.

Sonuçlar en uzun süre çalışan kişiye göre sıralanarak en üstteki kişi seçilir.

**Sorgu :**

SELECT TOP 1

e.BusinessEntityID,

p.FirstName + ' ' + p.LastName AS FullName,

d.Name AS DepartmentName,

DATEDIFF(DAY, edh.StartDate, ISNULL(edh.EndDate, GETDATE())) AS DaysInDepartment

FROM HumanResources.EmployeeDepartmentHistory edh

JOIN HumanResources.Employee e ON edh.BusinessEntityID = e.BusinessEntityID

JOIN Person.Person p ON e.BusinessEntityID = p.BusinessEntityID

JOIN HumanResources.Department d ON edh.DepartmentID = d.DepartmentID

ORDER BY DaysInDepartment DESC;

---En Uzun Süre Çalışan Departman Lideri---

**2️. Birbirine En Yakın İki Tedarikçi**

**Purchasing.Vendor** tablosundaki tedarikçiler, **Person.Address** ve **Person.BusinessEntityAddress** tabloları ile ilişkilendirilerek konum bilgileri (şehir ve posta kodu) alınır.

**PostalCode** baz alınarak aynı bölgede bulunan iki tedarikçi belirlenir.

**ORDER BY** ile sıralama yapılarak en yakın iki tedarikçi eşleştirilir.

**Sorgu:**

SELECT TOP 1

v1.Name AS Vendor1,

v2.Name AS Vendor2,

a1.City AS City,

a1.PostalCode AS PostalCode

FROM Purchasing.Vendor v1

JOIN Person.BusinessEntityAddress bea1 ON v1.BusinessEntityID = bea1.BusinessEntityID

JOIN Person.Address a1 ON bea1.AddressID = a1.AddressID

JOIN Purchasing.Vendor v2 ON v1.BusinessEntityID < v2.BusinessEntityID

JOIN Person.BusinessEntityAddress bea2 ON v2.BusinessEntityID = bea2.BusinessEntityID

JOIN Person.Address a2 ON bea2.AddressID = a2.AddressID

WHERE a1.PostalCode = a2.PostalCode -- Aynı posta koduna sahip olanlar

ORDER BY v1.Name;

---Birbirine En Yakın İki Tedarikçi---

**3️. Departmanlar Arasında En Çok Göç Eden Çalışan**

**EmployeeDepartmentHistory** tablosundaki **DepartmentID** değişiklikleri sayılarak bir çalışanın kaç farklı departmanda çalıştığı bulunur.

**COUNT()** fonksiyonu kullanılarak çalışanın toplam departman değişim sayısı hesaplanır.

**ORDER BY** ile en çok departman değiştiren çalışan en üst sıraya yerleştirilir.

**Sorgu:**

SELECT TOP 1

e.BusinessEntityID,

p.FirstName + ' ' + p.LastName AS FullName,

COUNT(edh.DepartmentID) AS DepartmentChanges

FROM HumanResources.EmployeeDepartmentHistory edh

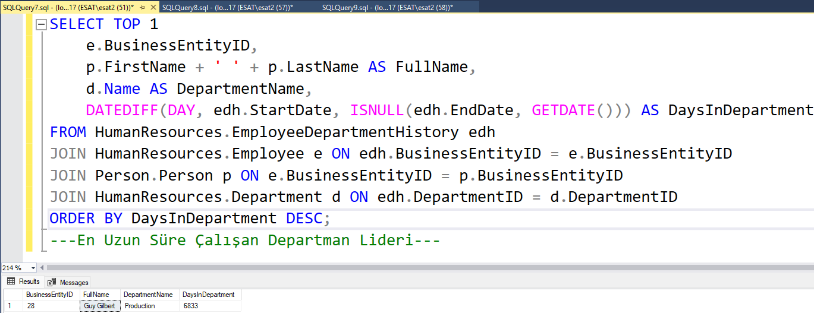
JOIN HumanResources.Employee e ON edh.BusinessEntityID = e.BusinessEntityID

JOIN Person.Person p ON e.BusinessEntityID = p.BusinessEntityID

GROUP BY e.BusinessEntityID, p.FirstName, p.LastName

ORDER BY DepartmentChanges DESC;

---Departmanlar Arası En Çok Görev Alan , Değiştiren Çalışan---

Sorgularımın Çalıştığına Dair Görseller: 

metin, ekran görüntüsü, yazılım, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

metin, ekran görüntüsü, yazılım, web sayfası içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.