**Lazy Loading vs Eager Loading**

Entity framework’ün yazılan LINQ sorgularına bağlı olarak veriyi veri tabanından yüklemek için kullandığı loading yöntemleridir.

# Lazy Loading

Lazy Loading, sayfada yer alan bir ögenin ihtiyaç duyulmadığı takdirde çağrılmaması prensibi ile çalışır yani bir nesne örneğinin gerçekten ihtiyaç duyulacağı ana kadar alınmaması ve bekletilmesi amacıyla kullanılır. Bu yöntemde veriler sorguya bağlı olarak çekilir ve veri setinin içindeki tüm dataları yüklemek yerine kullanılacağı an tekrar sorgu atar ve veriyi çeker.

## Örnek

Örneğin, aşağıda verilen sorguyu çalıştırdığımızda, UserDetails tablosu Kullanıcı tablosu ile birlikte yüklenmeyecektir.

User usr = dbContext.Users.FirstOrDefault(a => a.UserId == userId);

Yalnızca, aşağıda gösterildiği gibi, açıkça aradığınızda yüklenir.

UserDeatils ud = usr.UserDetails; // UserDetails are loaded here

Buda her yeni yükleme için veritabanına bir sorgu atılması demektir.

# Eager Loading

Lazy Loading’in tam tersi yönde çalışır. Kullanacağımız nesneleri, nesnenin ihtiyaç anından çok önce yaratır ve bekletir. Eager loading Linq sorgusu çalıştırıldığında verilerin tamamını yükler ve hafızaya alır.

## Örnek

Örneğin, bir Kullanıcı tablonuz ve bir KullanıcıDetaylar tablonuz (Kullanıcı tablosuyla ilişkili varlık), o zaman aşağıda verilen kodu yazacaksınız. Burada, kullanıcıyı kullanıcı detayları ile birlikte kullanıcı kimliğine eşit bir ID ile yüklüyoruz.

User usr = dbContext.Users  
 .Include(a => a.UserDetails)  
 .FirstOrDefault(a => a.UserId == userId);

Eğer birden fazla alt öğe seviyemiz varsa, aşağıdaki sorguyu kullanarak yükleyebilirsiniz.

User usr = dbContext.Users  
.Include(a => a.UserDetails.Select(ud => ud.Address))  
.FirstOrDefault(a => a.UserId == userId);

Bu, ilişkili varlıkları sorgunun bir parçası olarak döndüren ve bir kerede büyük miktarda veri yükleyen yöntem bize zaman ve hız kazandırıyor.

# Avantaj/Dezavantajları

İki farklı yönteminde kendine göre avantajı ve dezavantajı vardır bunlar geliştirmekte olduğumuz uygulamanın ihtiyaçlarına göre veri tabanımızın büyüklüğüne göre ve oradaki ilişkilerin çokluğuna göre değişir.

Yukarıdaki basit örnekten yola çıkarsak, lazy loading ile birbiriyle ilişkili olan entityler ihtiyaç oldukça çekilir. Bu da bize içinde bulunduğumuz case’e göre performans açısından yarar sağlayabilir. Eager loading’e göre veritabanına çok daha fazla kez bağlanır ve sorgu atar bunun da program için bir maliyeti vardır. Eager loading ise tek sorguda gerekli bilgileri elde eder.

Lazy loading ve Eager loading arasındaki çalışma hızı farkını değerlendirecek olursak, lazy loadingin tekrar tekrar database’e bağlanması sebebiyle hızı kayıt sayısı arttıkça eager loadingin üzerine çıkıyor.

## Lazy Loading Eager Loading

100 kayıt için ortalama 1 sn - 1 sn

1000 kayıt için ortalama 2.5 sn -2 sn

5000 kayıt için ortalama 7.5 sn -5 sn

10000 kayıt için ortalama 9.5 sn - 6.3 sn

KAYNAK: <https://medium.com/@e.karabudak7/lazy-loading-vs-eager-loading-67d09c6a251>