**Clustred Index**

*Clustred index , veriyi sql’de fiziksel olarak sıraya sokan yapıdır.* **Aslında hepimiz clustred index’i tablolarımızda kullanıyoruz.**Tablolarımıza tanımladığımız her bir**Primary key** aslında otomatik olarak bir **Clustred index yapısıdır**. Çünkü tablolarımız bu pk’ya göre fiziksel olarak sıralanır.

Bir tabloya PRIMARY KEY tanımladığınızda ve o tabloda başka hiçbir clustered index yoksa, SQL Server **varsayılan olarak** PRIMARY KEY kolonu (veya kolonları) üzerinde bir Clustered Index oluşturur. Ancak, bir tabloda zaten bir clustered index varsa (başka bir kolonda veya kolon kombinasyonunda), PRIMARY KEY tanımladığınızda SQL Server otomatik olarak **non-clustered** bir index oluşturur. Yani PK, mutlaka clustered index olmak zorunda değildir, ama genellikle öyle başlar veya ayarlanır. PK bir kısıtlamadır (constraint), Clustered Index ise bir index türüdür (fiziksel düzenlemeyi belirler).

**Clustered index ile ilgili önemli noktalar**

* Her tabloda yalnızca 1 adet clustered index olabilir.
* Sql query sonucu sıralı dataları dönerken de clustered indexe göre aynı sırada döner.
* Tablodaki bir clustered index pk olabileceği gibi aynı zamanda *birden fazla kolonun birleşiminden oluşan bir yapı da olabili*r. Buna **composite clustered index**denir.

**Primary key – Clustered Index farkı nedir?Primary key**dediğimiz, tablodaki kaydın uniqueliğini garantileyen bir alandır ve kaydın kimliğidir. Tanımlandığı gibi de clustered index özelliği taşır.

**Clustered index,**bir veri yapısı, dataya daha hızlı ulaşmak için oluşturulmuş bir indexleme şeklidir.

**Genel olarak datanın fiziksel sıralamasını düzenleyerek dataya ulaşma süresini optimize etmeyi amaçlarken, keyler ise datanın uniqueliğini sağlar.**

**Non-Clustered Index**

Bir kolonu Non-clustered index olarak indexlediğinizde,**arka tarafta yeni bir tablo oluşur (**daha doğrusu **ayrı bir B-tree yapısı oluştrurur**) **ve bu yapı sizin indexlediğiniz kolona karşılık kolon adresini tutar.** Yani : SQL Server, non-clustered index için ana tablonun kendisi dışında, index anahtarı kolonlarını ve bu anahtarlara karşılık gelen pointer'ları içeren ayrı bir B-tree yapısı oluşturur. Pointer, verinin ana tabloda nerede bulunduğunu gösterir.

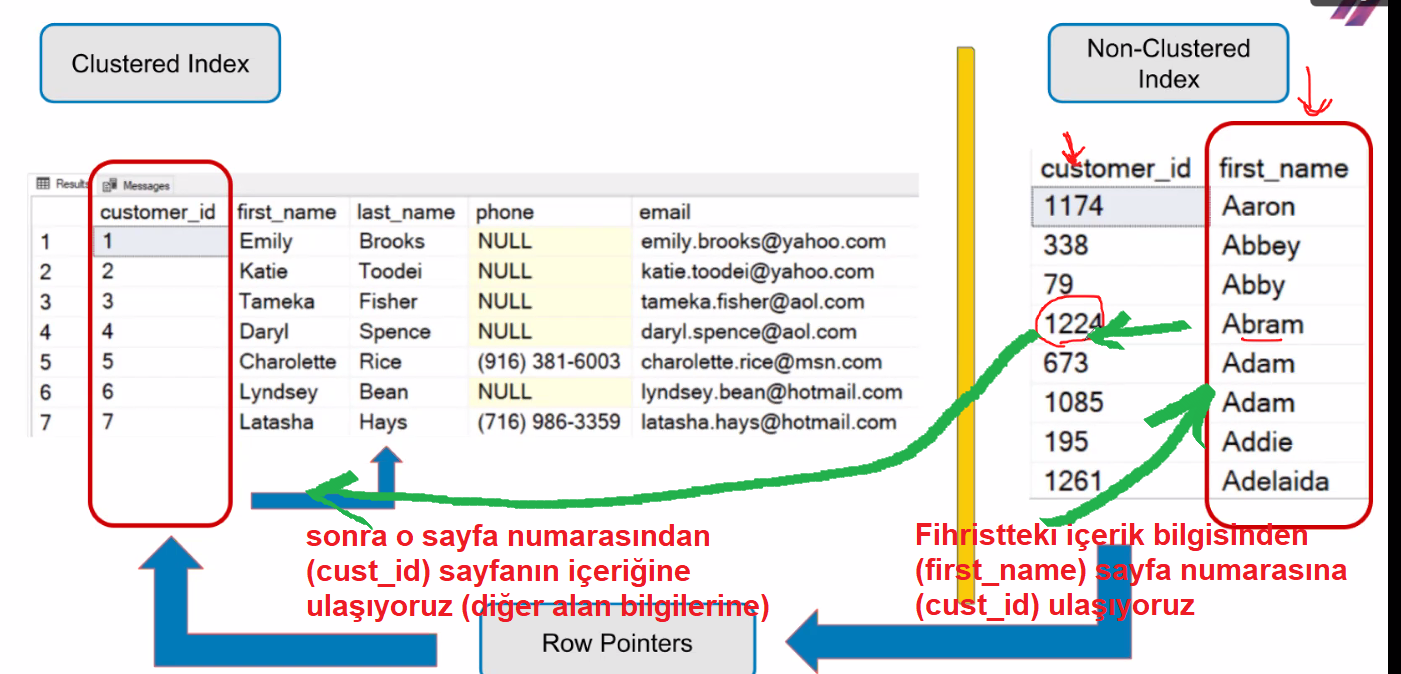
**Non-clustered indekste (NCI) verilere erişme:**

Veriye non-clustered index üzerinden erişilir, ancak bu erişim doğrudan verinin tamamına değil, önce index yapısına ve oradan da ana tablodaki veri satırına bir lookup (arama) adımı gerektirir. Yani erişim yolu NCI'dır, ama NCI'ın kendisi verinin tamamını içermez. Eğer sorgu sadece indexlenmiş kolonları veya INCLUDE edilen kolonları istiyorsa (Covering Index), bu durumda ana tabloya gitmeye gerek kalmaz ve erişim "daha direkt" (lookup olmadan) olabilir.

**Clustered index (CI) olan bir tabloda NCI varsa**: Pointer, o kaydın clustered index anahtar(lar)ıdır. Tam kaydı bulmak için önce NCI kullanılır, oradan elde edilen CI anahtarıyla ana tablodaki CI B-tree'si taranır ve kaydın tamamına ulaşılır. Yani elde edilen indeksleme yapısına erişmek için clustered indeks yapısı kullanılmış olur.

**Clustered index olmayan bir tabloda** (Heap) NCI varsa: Pointer, kaydın fiziksel adresi (RID - Row Identifier)'dir.

Kitapların başında içerik kısmı vardır. Bu içerik kısmında her bir konu başlığının hangi sayfa numarasında veya sayfa numaraları aralığında olduğunu gösterir. Siz kitabı açtığınızda önce içerik sayfasına bakarsınız. **Daha sonra aradığınız içeriğin sayfasını ya da sayfa aralığını öğrenip direkt olarak bu sayfalara geçersiniz**. Non-cluster index de tam olarak bunu yapmakta.

****