

یکی از مشکلاتی که با اکثر سیستم‌های رابطه‌ای وجود دارد، نیاز به تکمیل یک تراکنش برای دریافت Id رکورد ثبت شده است. همین مساله علاوه بر خاتمه یک تراکنش و شروع تراکنشی دیگر، به معنای رفت و برگشت اضافی به بانک اطلاعاتی نیز می‌باشد. به همین جهت در RavenDB برای ارائه راه حلی برای اینگونه مشکلات از الگوریتم HiLo برای تولید کلیدهای اسناد استفاده می‌گردد.

HiLo چیست؟

HiLo روشی است برای ارائه idهای عددی افزایش یابنده، جهت استفاده در محیط‌های چندکاربری. در این حالت، هنوز هم سرور تولید idها را کنترل می‌کند، اما هربار بازه‌ای از Idها را در اختیار کلاینت‌ها قرار می‌دهد. به این ترتیب کلاینت، بدون نیاز به رفت و برگشت اضافی به سرور، می‌تواند Id مورد نیاز خود را از بازه ارائه شده تامین کرده و هر زمانیکه این بازه به پایان رسید، سری دیگری را درخواست کند.

بدیهی است در این حالت، ارائه کلیه Idهای یک بازه از طرف سرور ممکن است کاری سنگین باشد. به همین جهت سرور در درخواست ابتدایی کلاینت، یک تک id را به نام «Hi» جهت مشخص سازی ابتدای بازه تولید idها، در اختیار او قرار می‌دهد. قسمت «Lo» توسط خود کلاینت مدیریت می‌شود و در این بین به هر کلاینت یک capacity یا تعداد مجاز idهایی را که می‌تواند تولید کند، از پیش اختصاص داده شده است:

```
(currentHi - 1)*capacity + (++currentLo)
```

بنابراین فرمول تولید idهای جدید در سمت کلاینت به نحو فوق خواهد بود. currentLo تا جایی افزایش می‌یابد که capacity آن کلاینت تنظیم شده است. سپس Hi جدیدی درخواست شده و Lo صفر می‌شود.

حالت پیش فرض کار کلاینت‌های RavenDB نیز به همین نحو است و الگوریتم HiLo در DocumentConvention آن، از قبل تنظیم شده است و مزیت مهم روش HiLo این است که با هربار فراخوانی متد Store یک سشن، بدون رفت و برگشت اضافی به سرور، Idهای لازم در اختیار شما قرار خواهد گرفت و نیازی نیست تا زمان SaveChanges صبر کرد. به این ترتیب batch operations در RavenDB کارایی بسیار بالایی خواهند یافت.