**SQL Server Policy Management ile isimlendirme kontrollü nasıl yapılır..**

Merhabalar,
Bugün sizlere SQL Server 2008 ile yaşantımıza girmiş bir özellikten bahsedeceğim. Bu özellik ile çeşitli policy oluşturup bunların takibini sağlayabilirsiniz. Örnek vermek gerekirse, tablo isimlendirmeleri hakkında bir policy oluşturup belirli aralıkta bunların kontrolünü tek bir noktadan sağlayabilirsiniz.

Policy Management menüsüne SQL Server Management Studio (SSMS) de management’ın altında erişebilirsiniz.



Yukarıda ki resimi incelemeye başladığımız taktirde Policy Management altında 3 bölüm olduğunu görüyoruz. Bu bölümler ve tanımlamalarını yapmamız gerekirse ;

* **Policies :** Condition( Koşul ) ların bütünleştirilip bir policy tanımadığımız bölüm.
* **Conditions :** İstediğimiz koşulları oluşturduğumuz bölüm.
* **Facets :** SQL Server kurulumu ile gelen yönetilebilen özelliklerin bulunduğu bölümdür. Bu bölüm içerisinde Table Facets’a bakarsak eğer bir tablo için koşul olarak verebileceğimiz bütün özellikleri görebiliriz. Aşağıda Table Facets örneği gösteriyorum.



Temel tanımlardan bahsettik,şimdi policy oluşturmanın basamaklarını beraber inceleyerek bir “Table Naming Policy” oluşturalım.

1. Condition ( koşul ) oluşturulması
2. Policy Oluşturulması
3. Policy çalıştırılması

Örneğimiz içerisinde oluşturulmasını istediğimiz policy, “DBA” veritabanı içerisinde “Table” isimlerini kontrol ederek “tbl\_” ön takısı ile başlayıp başlamadığını kontrol edecektir.

İşlem basamaklarını oluşturuyuz.

1. Öncelikli olarak “DBA” veritabanı oluşturuyoruz.

IF NOT EXISTS(select \* from sys.databases where name='DBA')

Create DATABASE DBA

1. Condition ( Koşul ) oluşturma işlemini gerçekleştirmemiz gerekiyor. Policy Management altında bulunan Condition üzerinde sağ tıklıyoruz ve new conditions tıklıyoruz.



Karşımıza yeni gelen pencere ise aşağıdaki gibidir. Bu pencere üzerindeki tanımlamar ise ;

* Name : Condition’a verilecek isim bilgisidir. Örneğimizde “**Table\_Name\_Condition**” olarak belirliyoruz.
* Facet : Condition’ı etkli olmasını istediğimiz alan bilgisidir. Örneğimizde “**Table**” olarak belirliyoruz.
* Expression : Bu bölüm ise kriterlerimizin tanımladığımız bölümdür. Örneğimizde Field bölümüne “**@Name**” operator bölümüne “**LIKE**” value bölümüne ise “**tbl\_%**” yazıyoruz.



1. Condition bilgisini de oluşturduktan sonra sıra policy oluşturmamıza geldi. Bu işlem için ise Management Policy altında bulunan Policies sağ tıklyıp new policies tıklıyoruz.



Gelen pencere ve üzerindeki değer bilgileri aşağıdaki gibi yapılandırılıp “ok” tıklanır.

* **Name :** Oluşturulan Policy için verilen isim bilgisidir. Örneğimizde verilen isim ise “**Table\_Name\_Policy**” dir.
* **Check Condition :** Policy içerisinde hangi condition bilgisini kriter olarak seçeceğimizi belirtiyoruz. Örneğimiz de bu bölümüde “**Table\_Name\_Condition**” seçiyoruz.
* **Against Targets :** Policy hangi hedeflerde gerçekleştirileceğiniz belirtiyoruz. Varsayılan olarak “**Every Table**”,”**Every Database**” gelmektedir. Bizim örneğimizde olması gereken ise “**DBA**” veritabanı içerisinde “**Every Table**” da olması. Bu yüzden Against Targets bölümünde “**Every Database**” bölümünde New Condition oluşturarak “**DBA**” veritabanı seçiyoruz. ( New Condition kısımı ikinci adımda gerçekleştirdiğimiz tanımlama benzeridir. Bu yüzden tekrar tanımlamanın nasıl yapılacağını anlatmadım.)
* **Evaluation Mode** : Bu bölümde karşınıza 2 adet seçenek gelmektedir. Bunlar ;
	+ **On Demond :** Default olarak gelmektedir. Bu seçeneğin seçilmesi halinda Policy üzerinde sağ tıklayıp evaluate seçeneğini seçip manuel çalıştırmanız gerekmektedir.
	+ **On Schedule :** Policy için bir zamanlama tanımlayıp o vakitlerde çalıştırılması sağlayabilirsiniz. Tanımlı zamanlama bilgilerini seçebileceğiniz gibi sizde tanımlama yapabilirsiniz.



Policy oluşturma işlemini gerçekleştirmiş olduk. Şimdi oluşturduğumuz “DBA” veritabanı üzerinde örnek iki adet tablo oluşturup policy evaluate edip sonuçları görüntüleyelim. Tablo oluşturmak için aşağıdaki kod parçalarını kullanabiliriz.

use DBA

go

create table MsHowto ( Id int)

go

create table tbl\_MsHowto ( Id int)

 Tablo oluşturma işlemini de başarılı bir şekilde yaptık şimdi policy evaluate etmemiz gerekiyor. Bu işlem için Policy Management 🡪 Policies 🡪 **Table\_Name\_Policy** üzerinde sağ tıklyıp evaluate ediyoruz.



 Evaluate işlemini yaptıktan sonra ;



Yukarıda Evaluate Policies ekranından anlaşılacağı üzere “**DBA**” veritabanı üzerinde Policy uygulandığı, 1 tablonun policy uygun diğerinin ise uygun olmadığı gibi bilgileri görebiliyoruz. Views bölümüne tıklayarak policy detaylarını condition ile birlikte görüntüleyebiliyoruz. Aynı zamanda bu ekranın çıktısını ise “**Export Results**” diyerek dışarı alabilmekteyiz.

Sonuç olarak, SQL Server Policy Management ile bir çok özellike hakkında policy uygulayabilir, bunları kontrol edebilir ve raporlarını yapabiliriz. Policy Management ile gerçekleştiremediğimiz işlemler için ise, yönetilebilir özelliklere ait trigger tanımlaması yaparak istediğimiz özelliklere sahip bir SQL Server hizmeti kullanabiliriz. Policy Management, çoklu database developer çalıştıran bir firma içerisinde database daha düzenli olmasını sağlayabiliriz. Bir sonraki makalemizde görüşmek üzere..

Çağlar Özenç
 Database Administrator