**Data:** Bilgisayarın alabildiği, işleyebildiği, sonuç üretebildiği ve saklayabildiği her şey veri olarak adlandırılır.

**Database:**

Bir database (veri tabanı), information (bilgiler) topluluğudur, birbirleriyle ilişkili bilgilerin depolandığı alandır. Bu bilgi çok **yapısallaştırılmış** bir şekilde muhafaza edilir. Yani veriler belli bir **sistematik** şekilde saklanır. Bilinen bu yapıyı çok iyi değerlendirerek bilgiye **hızlı** ve **doğru** bir biçimde erişebilir ve değiştirebiliriz.

İçinde bulunduğumuz bilgi çağında database‘ler hemen her yerdedir. Telefonumuzdaki rehber bile bir veri tabanıdır.

\* Kitapçıya gittiğinizde ve bilgisayarda belirli bir kitabı aradığınızda kitapçının kitap database’ine erişiyorsunuz demektir.

\* Eğer internet üzerinden bir ürünü “online” olarak satın alıyorsanız, Web Sitesi sahibi(satıcının) ürün database’ine ulaşıyorsunuz demektir.

\* Bankanız bütün parasal kayıtlarınızı kendi database’i üzerinde tutmaktadır. Aylık hesap ekstrelerinizi size gönderen bankanız aslında database-report /veri tabanı raporu bastırıp size göndermektedir.

\* Doktorunuza telefon edip randevu istediğinizde, sekreter, müsait zaman için aslında kendi database’ine bakmaktadır.

\* Arabanızı tamire götürdüğünüzde, servis teknisyeni, arabanız üzerinde eski tarihlerde ne türde işler yapıldığını görmek için kendi database’ine bakmaktadır.

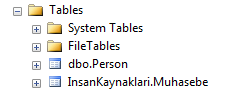
\* Bir süper markete gittiğinizde kasiyer, her ürünü barkod okuyucusu ile okuttuğunda, süper marketin database’inden o ürünün fiyatını aldığı gibi, stok kontrolünü de aynı zamanda birlikte yapmaktadır.

**Peki, neden veri tabanına ihtiyaç duyulmuştur? Bilgiler Excel ortamında da tutulamaz mıydı?**

Veri tabanı verilerin saklandığı, işlendiği ve belli bir kritere göre sorgulanarak kullanıcının hizmetine sunan bir sistemdir. Yani veri tabanları bize **büyük verileri okumak istediğimizde çok fazla kolaylık sağlıyor**. Örneğin **nüfus müdürlüğünde** o kadar kullanıcı arasından kendi bilgilerinize vakit kaybetmeden saniyeler içerisinde ulaşabiliyorsanız bu veri tabanı sayesinde olur. Aynı zamanda bir veri tabanında bir den fazla şart da yazılabilir sizin isminizde birden fazla kullanıcı var ise orada sizi ortaya çıkaracak harici bir sınırlayıcı ekleyerek Excel de yapılamayacak kullanım avantajları sağlıyor.(Alış-veriş sitesinde belli bir fiyata belli bir markaya ve belli özelliklerdeki ürünleri sınırlamamız gibi). **Veri tabanları, veri tabanı yönetim sistemleri aracılığıyla oluşturulur ve yönetilir**. **Microsoft SQL Server** bunlardan biridir.

Veritabanları **bilgi artışı** ile **birlikte bilgi depolama** ve **bilgiye erişim** konularında yeni yöntemlere ihtiyaç duyulması sebebiyle ortaya çıkmıştır.

***Schema nedir?***

**Şema** veri tabanında bulunan bütün nesneler (**tables**, **views**, **stored procedures**) schema içerisinde bulunur. **Bu nesnelere erişim schema üzerinden yapılır**. Böylece oluşturulmuş **nesneleri hem gruplandırmış hem de direk erişimi engellemiş oluruz**. (örn : HumanResources.Department)

<https://msdn.microsoft.com/en-us/library/ms189462.aspx>

**SQL (Structured Query Language - Yapılandırılmış Sorgulama Dili) :** Verileri ve veri kümelerini modellemek ve yönetmek için kullanılır. **Tanımlı anahtar kelimelerden oluşturulan SQL cümleleri** ile veritabanı üzerinde kayıt ekleme, silme ve var olan kayıtlar üzerinde listeleme yapabilir.

**Microsoft SQL Server**

MSSQL Microsoft firmasının oluşturduğu bir ilişkisel veri tabanı yönetim sistemidir. Çoğu zaman MYSQL ile karıştırılabilir. MYSQL açık kaynak kodlu bir **yazılım** iken MSSQL bu anlamda kullanıcılara maliyetli gelebilir. Web ya da masaüstü programlamada programcılar veri tabanı kullanacaklarsa eğer öncelikle bir veri tabanı sistemi seçmelidir. Bu ihtiyaçlara göre değişir. Bazı programcılar programdaki yapıları gereği Access’ i seçerken, bazıları da yine programın ihtiyaçları gereği MSSQL ya da MYSQL gibi veri tabanı sistemlerinden birini seçerler. Yani MSSQL aslında bir veri tabanı türüdür.

Bu programlar aracılığı ile MSSQL tipinde bir veri tabanı oluşturur, veri tabanını tasarlar, gerekli ayarları yaparsınız. Yani kısacası veri tabanı tasarımı yaparsınız.

Ancak aklınıza hemen şu gelebilir. **Microsoft Access de bir veri tabanı yönetim programıdır. Evet doğru o da bir veri tabanı yönetim programıdır, ancak MSSQL bir sunucu üzerinden çalışır. Yani oluşturduğunuz bir veri tabanı aslında sunucu üzerinde oluşturulur. Ancak Access öyle değildir. Access istemci yani kullanıcı tarafından oluşturulur ve taşınabilir bir veri tabanıdır.** MSSQL ‘ de ise veri tabanını sunucu üzerinde oluşturup, yine sunucu üzerinden tasarlarsınız. Merkezden kontrol esastır. Herkes ortak bir veri tabanına bağlanır ve veriler bu havuzdan kullanıcılara iletilir. Access bu anlamda biraz eksik kalıyor. Ayrıca daha stabil bir yapıya sahip olan MSSQL depoladığı veri açısından Microsoft Access’ e yine üstünlük sağlar.

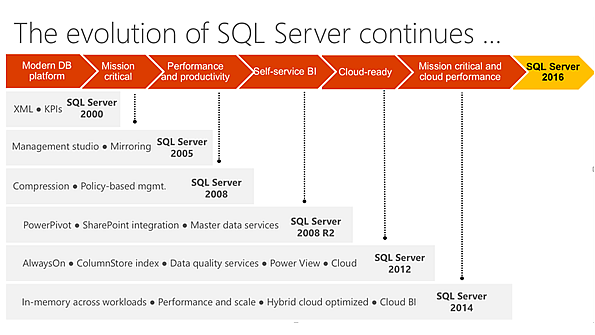
**Versiyonlar:**

**Temel Versiyonlar**

* Enterprise
* Standart
* Express
* Business Intelligence

**Temel Versiyonlar:**

* **Enterprise** 
  + En üst sürümdür.
  + Veri merkezleri için geliştirilmiştir.
  + Veri merkezi için gereken bileşenleri ve kritik iş yüklerinin yönetilmesi için servisleri içerir.
  + SQL Server kümelerini yaratmak ve düzenlemek için gerekli araçları içeren versiyon olup, çekirdek veritabanı motoru ve add-on servislerini desteklemektedir. **524 petabayta kadar veritabanı yönetebilmekte, 2 terabayt hafıza içermektedir**.
* **Business Intelligence**
  + Şirketler için geliştirilmiş sürümdür.
  + **Görselleştirme araçları ve tarayıcı üzerinden veriye erişim gibi özellikler içerir**.
  + 524 petabayta kadar veritabanı yönetebilmekte, 2 terabayt hafıza içermektedir.
* **Standard**
  + Küçük organizasyonların kullanımına uygundur.
  + Basit veri yönetimi, geliştirme araçları ve bulut desteği içerir.
  + SQL Server Standart versiyonu, çekirdek veritabanı motorunu, bağımsız servislerle (stand-alone services) birlikte içermektedir. Daha az aktif instance (kümedeki ağ sayısı) desteklemesi, hot-add memory (sunucu çalışırken hafıza eklenebilmesi) gibi yüksek süreklilik fonksyonları ve paralel indeksleri içermemesi, SQL Server Enterprise versiyonundan farklı olduğu noktalar olmaktadır. 524 petabayta kadar veritabanı yönetebilmekte, 2 terabayt hafıza içermekte ve örnek başına 128 GB maksimum bellek desteklemektedir.
  + **Express**
  + SQL Server Express versiyonu, çekirdek veritabanı motorunu içeren, ücretsiz bir versiyondur. Veritabanı ve kullanıcı sayısında bir sınırlama yoktur, ancak 1 işlemci, 1 GB hafıza ve 10 GB veritabanı dosya limiti bulunmaktadır.



**Sql Mirroring**

Felaketten kurtarma, iş  sürekliliği sağlamak adına Microsoft firması tarafından SQL Veritabanı Sunucuları için geliştirilmiş yüksek erişilebilirlik çözümlerinden bir tanesidir. SQL Server 2005 sp1 ile hayatımıza giren bu çözüm sonraki SQL Server versiyonları olan SQL Server 2008, SQL Server 2012 versiyonları tarafından da desteklenmektedir. **Bu yapıdaki amaç bir veritabanının iki farklı sunucuda kopyası oluşturularak, mevcut aktif sunucunun herhangi bir nedenden dolayı servis verememesi, devre dışı kalması ya da veritabanının bozulması durumunda yedek sunucu üzerinde bulunan veritabanı ile erişimin ve servisin aynen devam ettirilmesidir.**

**Principal :** Aktif olarak hizmet veren **kaynak veritabanının tutulduğu sunucudur**. Kaynak sunucu. Bu sunucu üzerinde bulunan veritabanının kendisi de **kaynak veritabanı veya principal database** olarak isimlendirilir.

**Mirror :** **Kaynak veritabanının kopyasının oluşturulduğu yedek sunucudur**. Bu sunucu üzerinde oluşan veritabanı da kopya veritabanı, yedek veritabanı veya **mirror database** olarak adlandırılır.

**Witness :** **Principal** ve **Mirror** sunucuları sürekli izleyerek bir **kesinti olayı gerçekleşmesi durumunda otomatik olarak rollerin değiştirilmesi görevini yapar.** Farklı bir tabirle database mirroring çözümünde ana sunucuda oluşabilecek bir kesinti durumunda yedek sunucuyu ana sunucu konumuna getirip manuel bir müdahaleye gereksinim olmadan sistemin aksamadan yürümesini sağlar. Witness sunucu kullanmak isteğe bağlıdır. Eğer otomatik geçiş istenmiyor veya buna bir sunucu ayrılamıyor ise kesintilerde rol geçişini manuel yapılacağı durumlarda tercih edilir.

Database Mirroring yapısna geçiş yapmadan önce dikkat edilmesi gerekenler;

-          Database Mirroring yapısına geçiş için karar verilecek olunursa Sql Server versiyonları minimum SQL 2005 SP1 olmalıdır.

-          Sql Server Sunucularının kurulu oldukları işletim sistemleri farklı olabilir. (Örn: Server 2008, Server 2012)

-          Sql server kurulu olan işletim sistemleri 32 bit veya farklı olarak 64 bit olabilir.

-          Principall ve mirror sunucuları aynı versiyon ve sürümde olmaları zorundadır.(Enterprise, Standart)

-          Database Mirroring Enterperise ve Standart versiyonlar tarafından desteklenmektedir.

-          Witness sunucusu ise Sql 2005 SP1 üzeri bir sql versiyonu olabilir. Express Edition bile kullanılabilir.

**Self-Service BI nedir?**

**Self-Service BI** bir nevi kendin pişir kendin ye mantığında iş zekâsıdır. **Özetle bu kapsamda, anlık ve sunum odaklı iş zekâsı ihtiyacı,** **teknik yeterlilik beklenmeyen** bilgi çalışanları(**Information Worker**) tarafından kolayca adapte olunabilecek araçlar vasıtasıyla üretilebilmektedir. Bu sayede teknik ekibe ihtiyaç azalmış ve sonuç itibariyle hem bilgi çalışanları hem de teknik ekip **maksimum fayda** sağlanabilecek konulara odaklanabilir olmuştur.

Microsoft bu alandaki ürünlerini yıllar öncesinden müşterilerine çeşitli şekillerde sunmaktaydı. **Şimdiler ise neredeyse tüm BI ihtiyacını karşılayabilen çok daha yetenekli yeni ürünlerini Power BI adı altında bir araya getirmektedir.**

Son durumda **Power BI kapsamında birbiri ile ilişkili dört üründen bahsedebiliriz**.

Bunlar ürünler şunlardır;

**Excel:**Excel 2013 içinPowerPivot, PowerView, PowerQuery ve PowerMap eklentileri mevcut iken Excel 2010 için sadece PowerPivot ve PowerQuery eklentileri mevcuttur. **(Excel’de hücrelere veri doldurmadan analiz yapabiliyorsun-online database’den inceleme yapılabiliyor)**

**Power BI for Office 365:** Office 365 üzerindeki portal hizmeti **SharePoint Online** olarak isimlendirilir. SharePoint kadar yetenekli olmasa da ortalama ihtiyaçları karşılayabilen bu ürüne SharePoint Store’dan indirilen Power BI bileşeni entegre edilebilmektedir. Office 365’e bağlanıp PowerView raporlarını **Mobile App**. üzerinden mobile cihazlarda sunabiliriz.AyrıcaPower BI sayesinde raporları **HTML5**’e dönüştürerek diğer cihazlarda da görüntülenmesini sağlayabiliriz. Ek olarak **Q&A** özelliği ile doğal cümlelere cevaben veri modellerimizden anlık raporlar elde edebiliriz.

**SharePoint:**Burada Excel dosyalarını paylaşabilir, versiyonlama, zaman bağlı olarak güncelleme gibi daha birçok özelliği kullanabiliriz. PowerPivot ve PowerView odaklı çalışan **PowerPivot Galeri** bu ürün dahilinde kullanılabilmektedir. Excel içindeki PowerView raporlarını görüntülemeye ek olarak, web ara yüzü ile PowerView raporları oluşturulabilir ve PowerPoint sunumu olarak çıktı alabiliriz**. (Kurumsal portal oluşturmak için)**

**Power BI Mobile App.:** Windows Store’dan indirilebilen bu uygulama ile Office 365’e bağlanabilir, buradaki raporları mobil cihazlarda veya Windows işletim sistemli bilgisayarlarda görüntüleyebiliriz.

**AlwaysOn:** AlwaysOn teknolojisi ile database’leriniz sürekli erişilebilir hale gelmekle birlikte, read-only kopyalar üzerinde maliyetli sorgularınızı çalıştırıp performans kazanabilir, backup maliyetlerinizi düşürebilirsiniz. Mirroring den farklı olarak birden fazla database’i gruplayarak bir sql instance’i üzerindeki birden fazla database’inizi sürekli erişilebilir (high availability) kılabilirsiniz. **Bu durum özellikle sharepoint, biztalk gibi birden fazla database üzerinde işlem yapan uygulamalar için çok faydalı olacaktır**. (Birden cok SQL Server kurarak birbiri ile mirroring yaptırabilirsin. Biri master db olabilir, diğer kopyasi mirror olan read modda açarak, okuma yapacaklari oraya yönlendirebilir ve okuma yapanlarin kayit yapanlari yavaslatmasi engellenebilir)

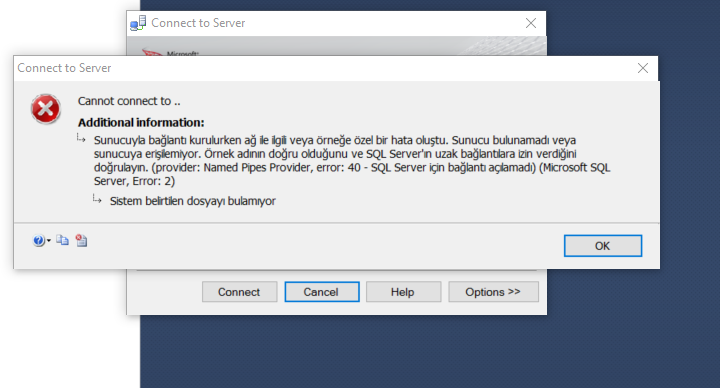
<http://ismailadar.com/sql-server-always-on-mimarisi/>

**SQL Management Studio**

SQL Server Management Studio, **SQL Server'ın tüm bileşenlerine erişmeye**, **bu bileşenleri yapılandırmaya**, **yönetmeye ve geliştirmeye yönelik** **tümleşik bir ortamdır**. SQL Server Management Studio, tüm beceri düzeylerindeki geliştiricilerin ve yöneticilerin SQL Server'a erişmesini sağlamak için çok çeşitli grafik araçlarıyla bir dizi zengin komut dosyası düzenleyicisini bir araya getirir.

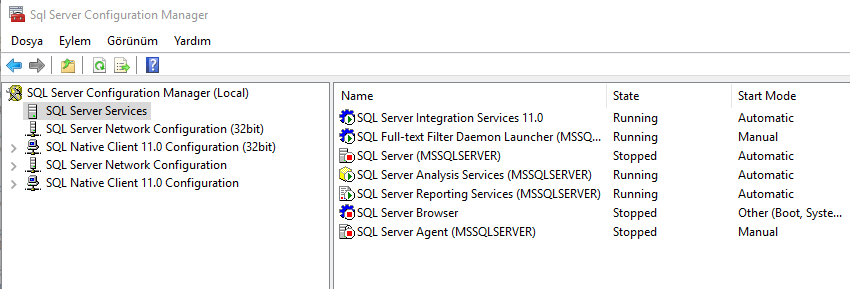
SQL Server Management Studio SQL Server'ın daha önceki sürümlerinde yer alan Enterprise Manager, Query Analyzer ve Analysis Manager özelliklerini tek bir ortam içinde bir araya getirir. Ayrıca SQL Server Management Studio, SQL Server'ın Reporting Services ve Integration Services gibi tüm bileşenleriyle çalışır. Geliştiriciler tanıdıkları bir deneyim yaşarken, veritabanı yöneticileri, kullanımı kolay grafik araçları ile zengin komut dosyası yazma yeteneklerini bir araya getiren, tek ve kapsamlı bir yardımcı programa sahip olur.

**Microsoft SQL Server Management Studio Açılış Hatası**

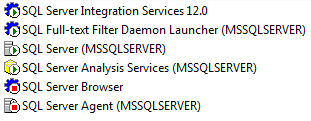


**Micsosoft SQL Server Management Studio ‘ yu Başlatmak – Duraklatmak - Durdurmak**

1. **Tüm programlar** -> **SQL Server Configuration Manager** tıklayın



1. Açılan penceren Sol kısımdan SQL Server Services kısmına geldiğinizde Sağ ksıımda SQL Server’ın servisleri yer alır.

**Servisler:**

**SQL Server** **Integration Services :** Örneğin, İzmir Şubemiz veri giriyor sürekli bizde izmirden gelen verilerin üzerinde bir işlem yapıp ana server ımızda bi yerde saklamak istiyorsak**SSIS(ETL)** servisimizin açık olması gereklidir. **ETL= Extract-Transform-Load.**Bir kaynaktan veriyi al dönüşümünü yap nerede saklayacaksan oraya yükle.

**SQL Server Full-text Search** : Kendi içimizde veri tabanımızda bir arama yapıyoruz ve verdiğimiz kriterlerde arama bulamazsa Server, ona benzer en yakın sonuçları getirsin istiyorsak bu servisi kullanırız.**SQL Server Full TEXT .** aradığınız sonuç bulunamadı diyip aramayı bitirmek yerine aradığınız sonuç bulunamadı , benzerleri aşağıdaki gibidir. Ya da bunu mu demek istediniz  yazılması için gerekli olan servistir. Tabii sadece yazmak için değil. Aranılan sonucu bulamadığı an ona en yakın kriterdeki sonuçları getirir.

**SQL Server (MSSQLSERVER)** : Sağ tık 🡪 ‘stop’ veya ‘pause’ komutlarıyla servis açılıp kapatılabilir. Sql server management studio açılırken bağlantı problemi yaşıyorsak bu servisi kontrol etmemiz gerekir.

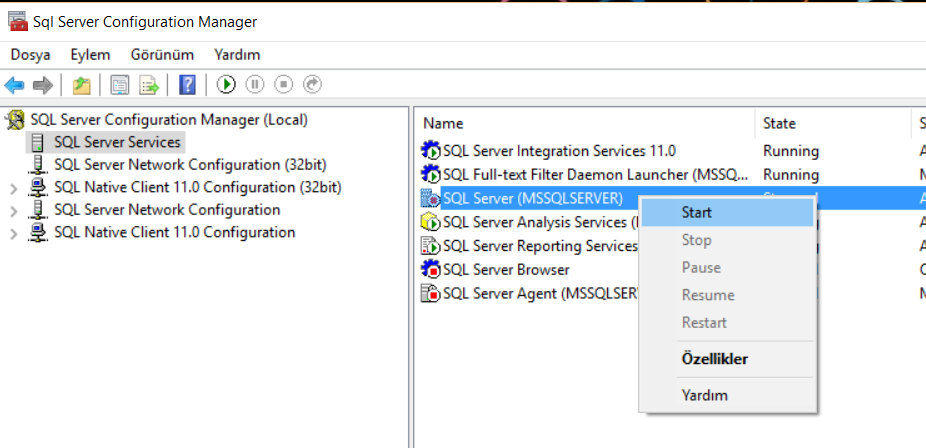
**SQL Server Analysis Services** : Veri üzerinde istatiski sonuçlar elde etmeye yarayan bir servistir.

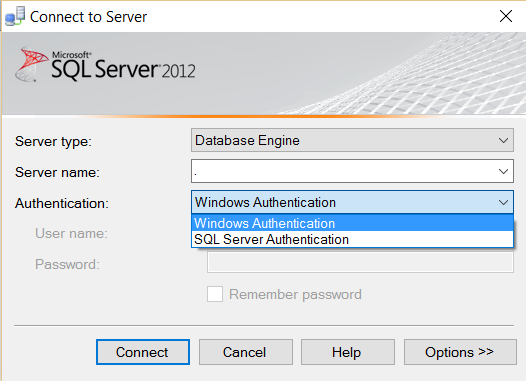
**SQL Server Agent** : Mesela şirket içi zamanlanmış görevler varsa ve bu görevleri ifa etmeleri konusunda  zamanında çalışanları uyarmak gerekiyorsa SSAgentS  i kullanmamız gerekli **( SQL SERVER  AGENT SERVICE ).**

**SQL Server Browser** : Kendi ağımızda server ımızın browser yardımıyla bulunmasını istersek te SSBS servisi(sQL Server Browser Service) nin açık olması gerekli.

**SQL Server Reporting Services** : Sonrasında da Patron yapılan tüm satışların v.s raporunu isterse  **SSRS=SQL SERVER REPORTING SERVICE**  in açık olması gerekir yapılan tüm veri manipılasyonları hakkında Rapor verilmesini sağlar.

1. Sağ ksıımda yer alan SQL Server (MSSQLSERVER) servisini Stopped’ den Running’e çevirdiğimiz anda Mangement Studio açılır hala gelecektir.



**Giriş Ekranı**

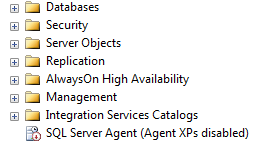
**Server Name:** localhost , ‘.’, veya kendi makinemizden bağlanıyorsak makinenin ismi olabilir.

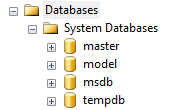
Sql Server ile clientlar arasında iki çeşit kimlik doğrulama yöntemi vardır. Windows Authentication ve Sql Server Authentication. Sql Server kurulumunda hangi yöntem ile kimlik doğrulama yapacağınız size sorulur.

**1- Windows Authentication :** Bu şekilde giriş yapılırken Windows kullanıcısı ile login olunur.

**2- Sql Authentication :** Burada ise Sql Server’a daha önceden belirlenmiş bir Kullanıcı Adı ve Şifre ile bağlanılır.

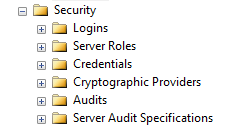
**Not:**// Windows kimlik doğrulama yöntemi seçilirse SQL Server’a sadece Windows tarafında kullanıcı hesabı olan ve aynı zamanda SQL Server tarafında da yetki verilmiş hesaplarla bağlanılabilir. Yani kullanıcı için önce Windows tarafında bir kullanıcı hesabı açılmalı ve bu hesabın kendisi ya da üye olduğu bir grubu SQL Server’a tanımlanarak erişim yetkisi verilmelidir. Bu seçenek seçildiğinde SQL Server için kimlik doğrulaması Windows tarafından gerçekleştirilir. Ve SQL Server kimlik doğrulamasında Windows’a güvenir. Bundan dolayı bu yöntemle SQL Server’a yapılan bağlantılara Güvenilen Bağlantı (Trusted Connection) adı verilir.



**System Databases;**

**\*master** : Sql’in tüm ayarlarını barındıran database’dir. Ana database. **\*model** : Her database’in kopyasını taşır. **\*msdb** : Zamanlanmış görevlerin database’idir. **\*tmpdb** : Geçici database.

**Yeni kullanıcı tanımlamak;**

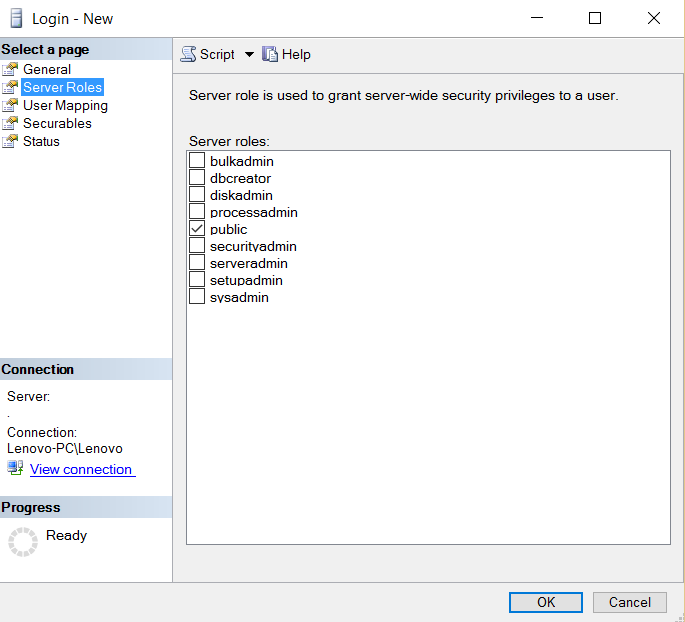
**\*logins** : Bağlanan kullanıcılar.

Sağ tık 🡪 New Login -- General

Login Name yazılır.

SQL Server authentication seçilir ve şifre belirlenir.

Kullanıcının yetkilerini belirlemek;



Bir Sql Server hesabı varsayılan olarak hiçbir yönetici yetkisi olmadan oluşturulur. Bu role public rolü denir. Daha sonra hangi veritabanına erişmek istiyor ise **Logins – Properties – User Mapping** sekmesinden veritabanı yetkileri atanır. Eğer bir kullanıcıya yönetici yetkileri vermek istiyorsanız, Login’nin server roles sekmesinden gerekli rolü verebilirsiniz.

Sql Server’da roller aşağıdaki gibidir;

-          **Bulkadmin**: Bulk insert yapma yetkisine sahiptir. Bir dosyadan veritabanına kayıt eklemek için kullanılır. Genellikle excelden veri çekmek için kullanılmaktadır.

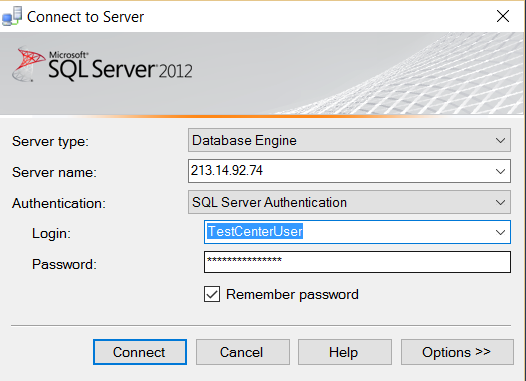
-          **Dbcreator**: Herhangi bir veritabanını oluşturma, düzenleme ve kaldırma yetkisi vardır. Yazılımcılara test amaçlı bu yetkiden verilebilir.

-          **Processadmin**: Sql proseslerini görme ve sonlandırma yetkisi vardır. Tüm çalışan prosesleri görebilirler. Kill komutu ile istenen proses veya prosesleri sonlandırabilirler.

-          **Securityadmin**: Sql hesabı oluşturup silebilirler. Server rolü oluşturma yetkileri yoktur.

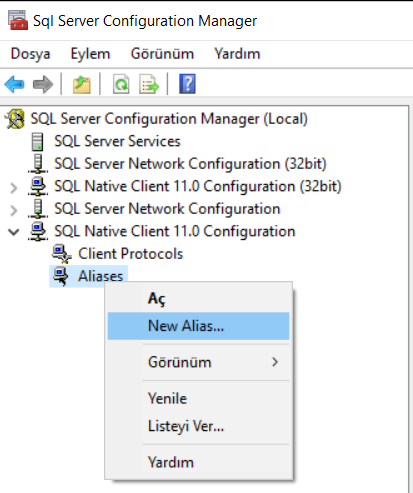
-          **Serveradmin**: Sql server instance’ının özelliklerini değiştirebilir, yeniden başlatabilir veya hizmeti durdurabilirler.

-          **Setupadmin**: Linked server (**SQL Server’in icine baska bir SQL Server’in baglantisini tanimliyorsun boylelikle obur SQL Server’in tablolarini kullanabiliyor**) oluşturma yetkileri vardır.

****-          **Sysadmin**: Tüm yönetici yetkilerine sahip roldür.

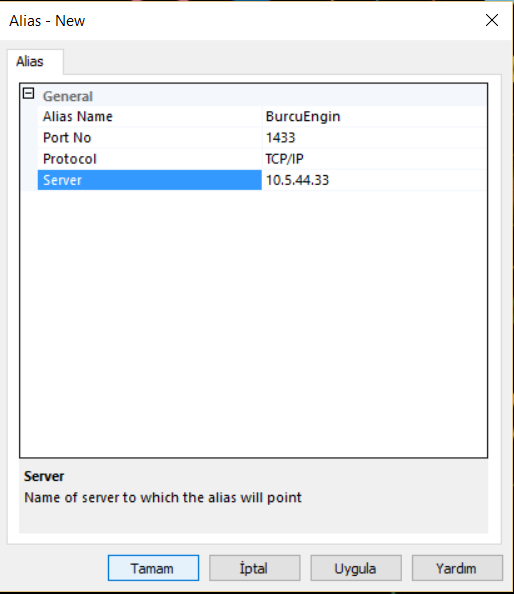
**SQL Server Management Studio’ya da Farklı Bir Bilgisayara IP Adresiyle Bağlanma**

**Not://** IP adresini öğrenmek için Komut penceresinden (Command prompt) **ipconfig** yazarak öğrenebilirim. Ama sadece şifreli üyeliklere girebilirim.

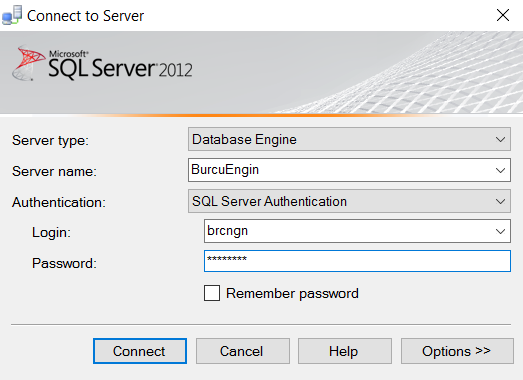


**SQL Server Management Studio’ya da Farklı Bir Bilgisayara İsimle Bağlanma**

SQL Server Configuration Manager’dan aşağıda yer alan sol taraftaki Aliases kısmına gelip Yeni bir Aliases (Takma Ad) ekleyelim



Tamam’ı tıkladıktan sonra tekra SQL Server Management Studio’ya penceresine gelerek, oluşturduğumuz takma adı deniyelim.



Conncet dedikten sonra bağlanması gerekmektedir.

**Database Files:**

SQL Server içinde iki çeşit dosya türü vardır:

* primary data file (birincil veri dosyası)
* secondary data file (ikincil veri dosyası)

**Birincil Veri Dosyaları (Primary Data File)**

Yeni bir veri tabanı oluşturduğumuz zaman, bütün bilgileri içermek için, tek bir primary (birincil) data file oluşur.

Birincil veri dosyaları (primary data files), varsayılan değer olarak \*.mdf uzantılı olarak verileri saklarlar.

**Secondary Data File (İkincil Veri Dosyası)**

Veri tabanı dosyalarının boyutunu genişletmek için, ikincil veri dosyaları (secondary data file) oluşturulur.

**Log Files (Log Dosyaları)**Veri tabanında bilgilerimizi saklarken hem yerden kazanmak isteriz, hem verilerimizi güvenli şekilde elde tutmak isteriz. Bilgilerimizi saklarken veri tabanı nesnelerini ve transaction log (işlem günlüğü) dosyalarını kullanırız. **Her bir veri tabanı en az bir tane veri dosyası ve bir tane log dosyası içerir**. **Log dosyası, veri tabanında meydana gelen değişiklikleri tutan transaction log dosyasını içerir**. Transaction log dosyaları, veri tabanında meydana gelen bütün değişiklikleri, veri tabanı için ayrılmış olan alanda saklar. Bütün düzeltme işlemleri, veri tabanına yazılmadan önce, transaction log dosyalarına yazılır**. Transaction log dosyaları veri tabanı aktivitelerini devamlı olarak kaydeder.** **Bu sayede, veri tabanın verilerimizi kayıp etmemize sebep olacak bir durum gerçekleştiğinde, bu dosyaları veri tabanı kurtarma işlemlerinde kullanabiliriz.**

**SQL Management Studio ile Veritabanı Oluşturma**

Sql Server Management Studio’ya giriş yaptıktan sonra “Databases” (veritabanları) üzerinde <sağ tuş> yaparak “New Database” seçilir.

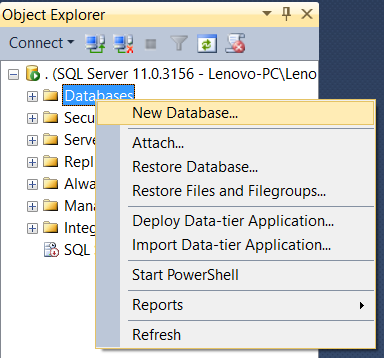
Veritabanına bir isim verilir, “Database name” ve <OK> butonuna basılarak yeni veritabanı oluşturulmuş olur. –

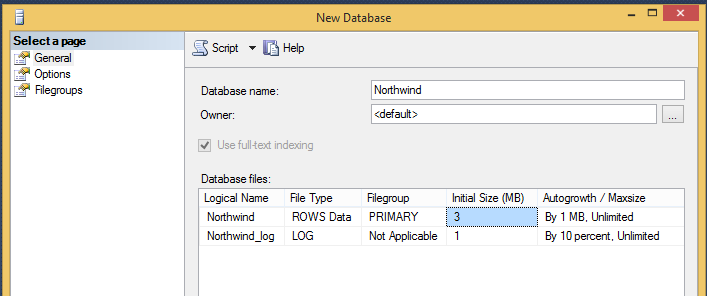
**Options incele**

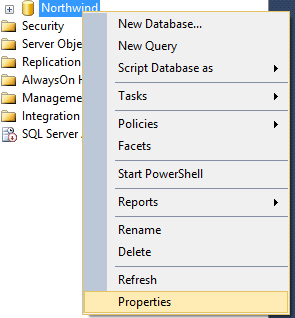
**CS-CI – Case Sensitive**

**AS-AI – Accent Sensitive**

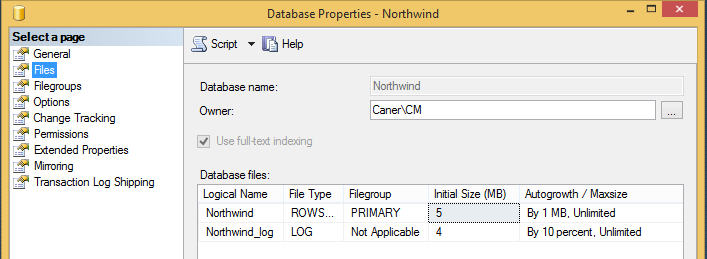
**Turkish\_CI\_AS**



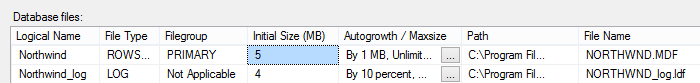
****

****

Veritabanı oluşturulduktan sonra <sağ tuş> <properties> seçeneği ile veritabanı özelliklerini kontrol edelim.

****

<Files> seçeneğini seçtiğimizde, veritabanı oluşturulduğunda “**Database Files” kısmında 2 farklı dosyanın oluşturulduğunu** görürüz. Genel anlamda bu dosyaların oluşturulma amaçları aşağıda belirtilmiştir:

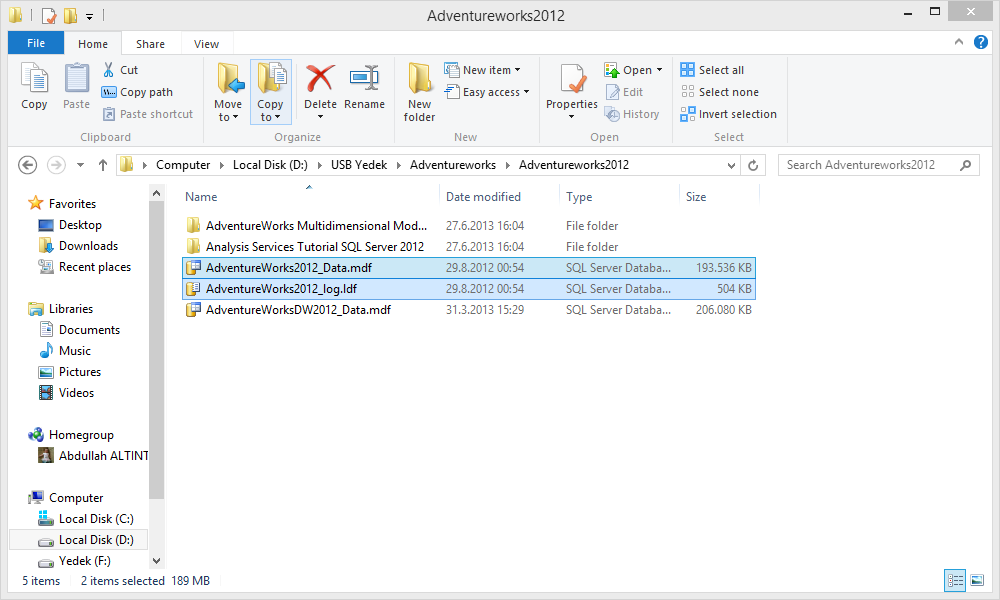
****

* **Yarattığımız database’i göremiyorsak Refresh (F5) yaparız. .ldf olmadan .mdf açılmaz !!! Bir database’e ait bu iki dosya olmak zorunda**.

**Attach-Detach**

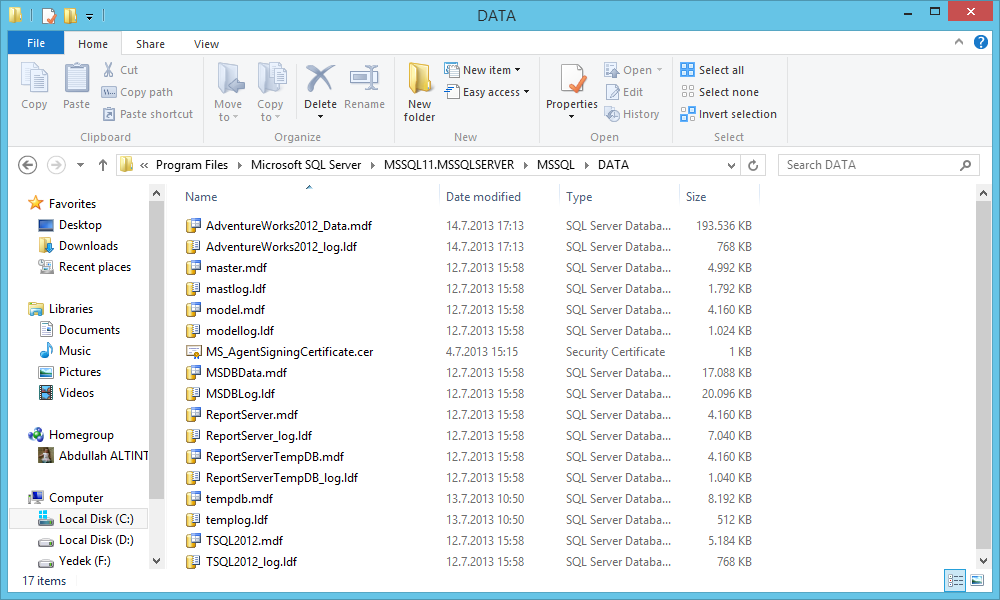
Microsoft SQL Server kullanarak veritabanlarının oluşturulması ve yönetimini kolay bir şekilde yapabilmekteyiz. Pratikte SQL Server kullanılarak yeni veritabanı oluşturulması ve bu veritabanı üzerinden işlemlerin yapılması kadar, daha önceden oluşturulmuş, içerisinde nesneleri barındıran hazır bir veritabanını SQL Server yönetimine dahil etmek de büyük bir ihtiyaç olarak karşımıza çıkmaktadır. Bu makalemizde daha önceden kendi local makinemizde veya başka bir makinede oluşturulmuş, içerisinde table, stored procedure, trigger, function, view vs. gibi database nesnelerini barındırabilen hazır bir veritabanını SQL Server instance ımızın kullanımına nasıl dahil edebileceğimizi ve artık kullanımını SQL Server instanceımızdan çıkarmak istediğimiz bir database i nasıl çıkarabileceğimizi inceleyeceğiz.

Öncelikle SQL Server yönetimine dahil etmek istediğimiz veritabanına ait Data (.mdf) ve varsa Log (.ldf) dosyalarını kopyalayalım. **(mdf dosyasinin alindigi SQL Server ile buradaki SQL Server versiyounun ayni olmasi gerekiyor)**

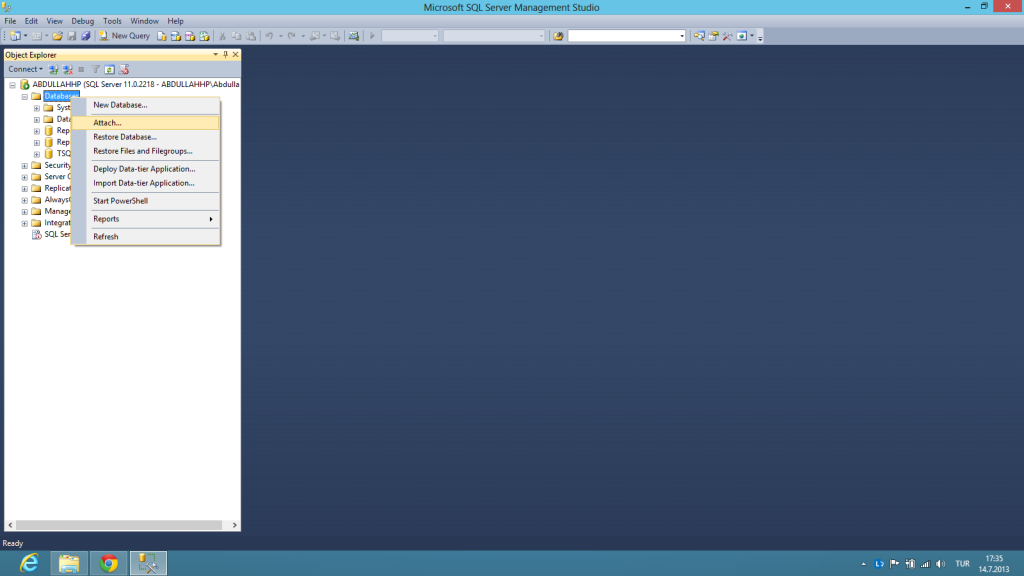
[](http://www.abdullahaltintas.com/wp-content/uploads/AdvFiles.png)

Örneğimizde ***AdventureWorks2012*** database’ine ait AdventureWorks2012.mdf ve AdventureWorks2012.ldf dosyalarını kopyaladık ve SQL Server’a ait DATA dosyalarının tutulduğu,

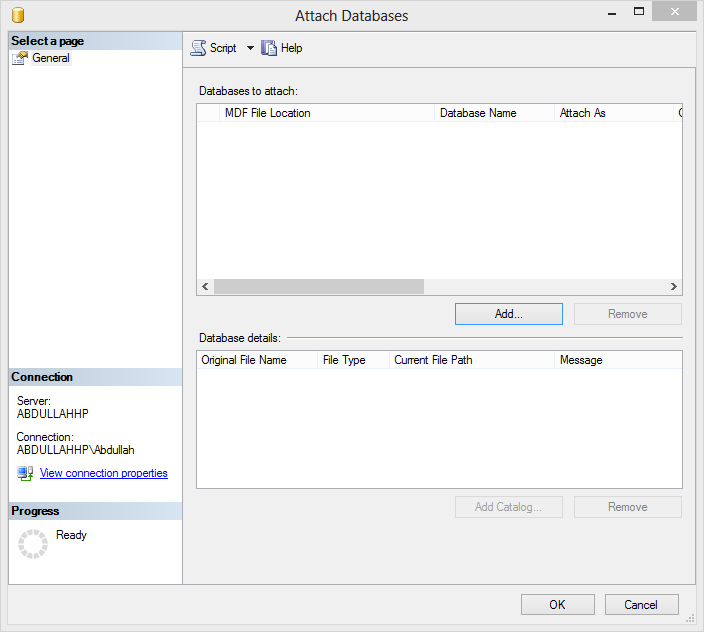
**C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA** klasörü altına yapıştırıyoruz.

[](http://www.abdullahaltintas.com/wp-content/uploads/DBPath.png)

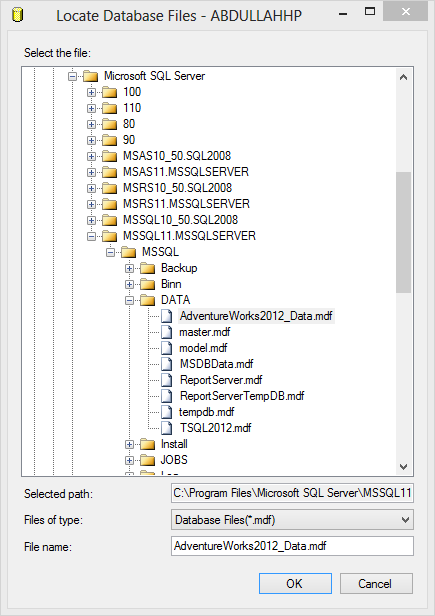
Bu işlemin ardından ***SQL Server Management Studio*** aracını açarak ilgili instance ımıza bağlanalım. Ardından sol tarafta bulunan **Object Explorer** bölümündeki **Databases**sekmesinin üzerinde sağ tıklayalım. Karşımıza çıkan menüden **Attach**bölümüne tıklayalım.

[](http://www.abdullahaltintas.com/wp-content/uploads/Screenshot-8.png)

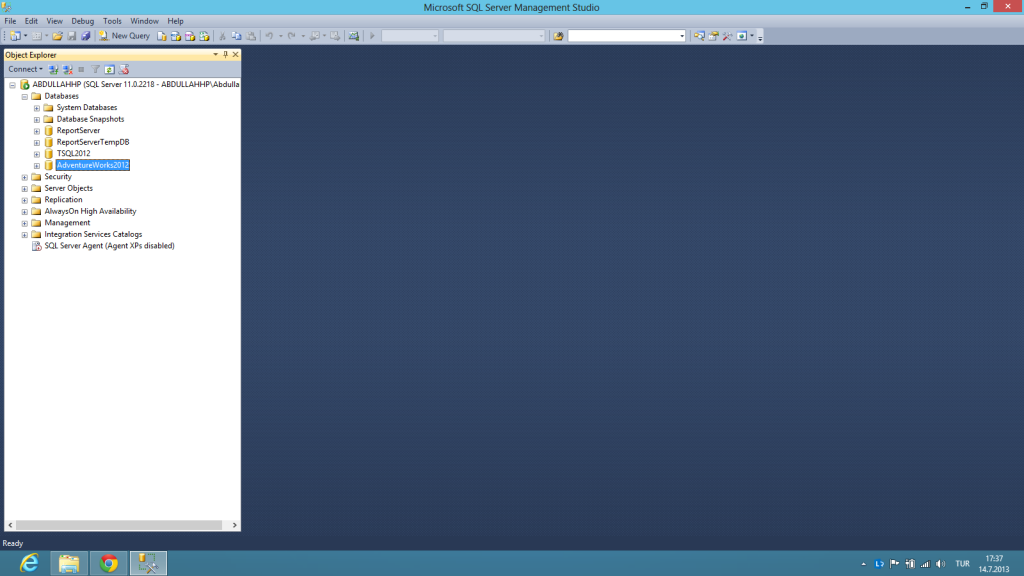
Karşımıza gelecek olan Attach Database ekranında ***Add***butonuna tıklayalım.

[](http://www.abdullahaltintas.com/wp-content/uploads/Attach1.png)

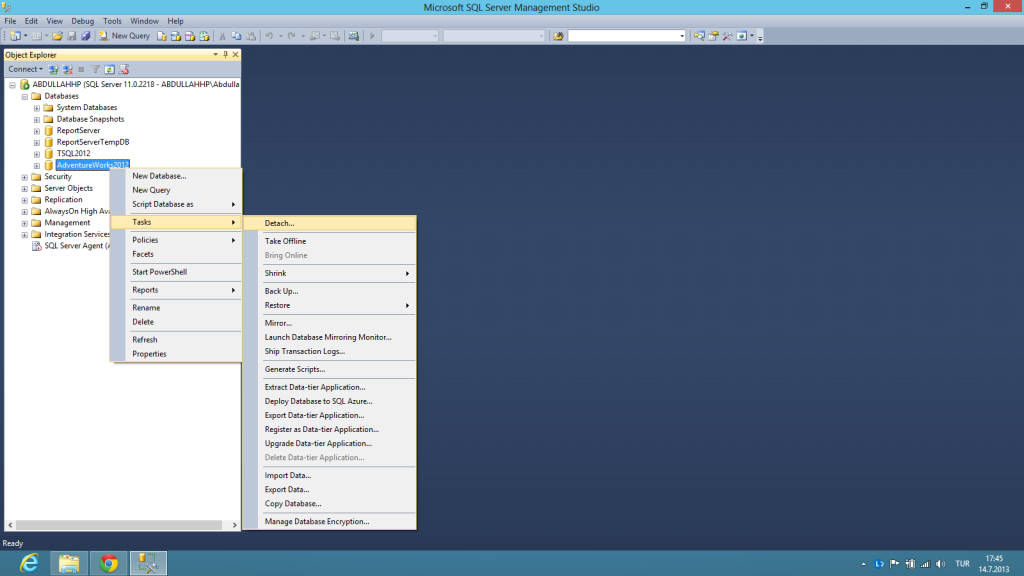
Az önce dosyalarını yapıştırdığımız AdventureWorks2012veritabanını seçelim ve ***OK***diyelim.

[](http://www.abdullahaltintas.com/wp-content/uploads/Attach2.png)

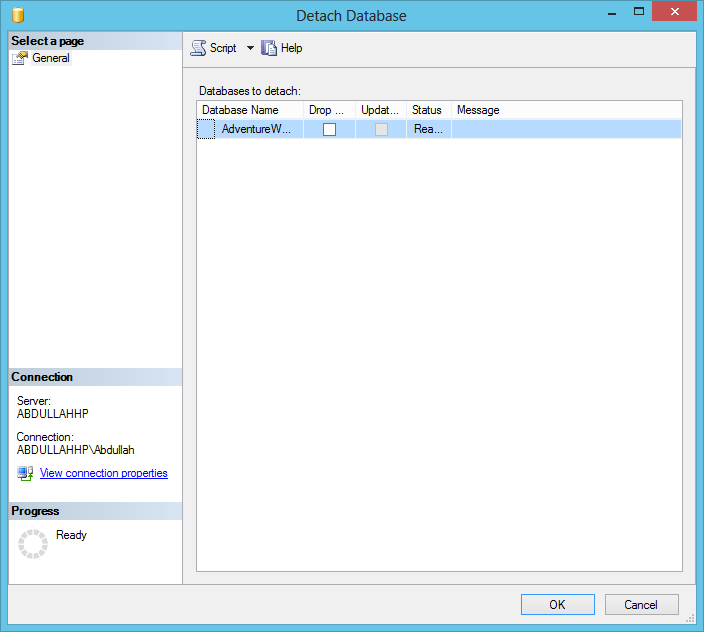
Bu işlemin ardından varsa log dosyasını SQL Server datadosyasına dahil edecektir, log dosyası yok ise SQL Server yeni bir log dosyası oluşturacaktır. ***OK***diyerek ekranda işimiz bittiğinde SQL Server yönetimine AdventureWorks2012database inin dahil edildiğini görebilirsiniz.

[](http://www.abdullahaltintas.com/wp-content/uploads/Screenshot-9.png)

**Son olarak SQL Server yönetiminde olan bir database in server kontrolünden nasıl çıkartılacağına bakalım**. Bu işlem için SQL Server yönetiminden ayırmak istediğimiz veritabanının üzerinde sağ tıklayarak ***Tasks –> Detach*** bölümüne tıklıyoruz.

[](http://www.abdullahaltintas.com/wp-content/uploads/Screenshot-10.png)

Karşımıza gelen ***Detach Database*** ekranında ***Ok***diyerek veritabanını SQL Server yönetiminden çıkartabiliyoruz.

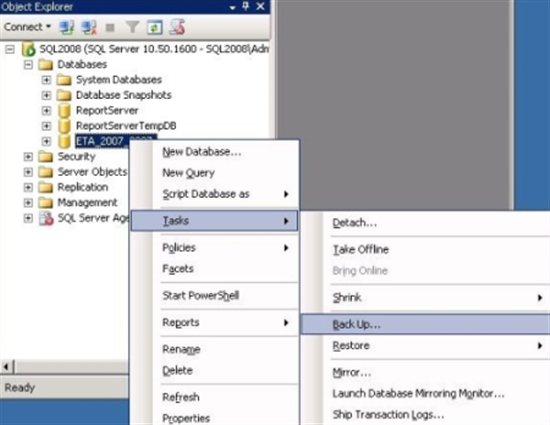
[](http://www.abdullahaltintas.com/wp-content/uploads/Detach.png)

Görüldüğü gibi ***Database Attach ve Detach*** işlemleri oldukça kolay bir şekilde grafik arayüz üzerinden gerçekleştirilebiliyor.

**SQL BACKUP**

Backup🡪Veritabanı yedeğini alma

Şekilde görüldüğü gibi  ilgili veritabanı üzerinde sağ buton > back up  diyoruz.

[](http://social.technet.microsoft.com/wiki/cfs-file.ashx/__key/communityserver-wikis-components-files/00-00-00-00-05/3240.001.jpg)

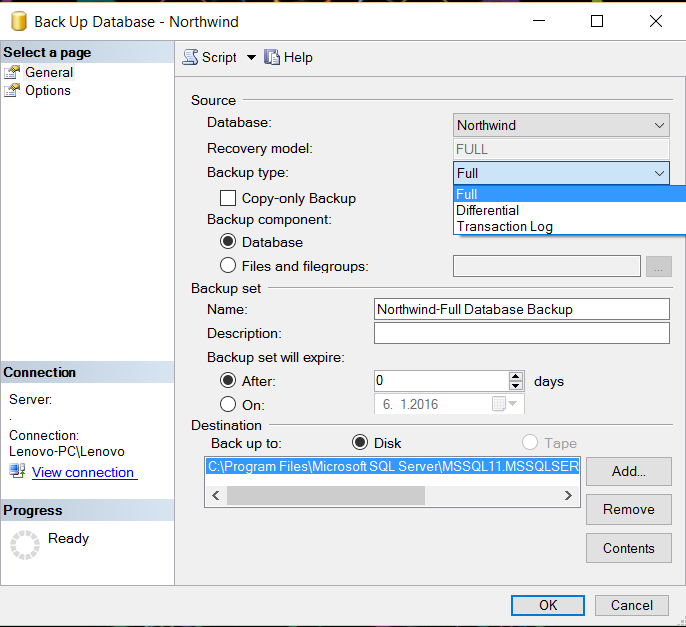
 Back Up Database ekranında alınacak backup için belirtmemiz gereken parametreler var. Burada yer alan önemli parametreler;

Backup Type : Alınacak yedeğin türünü belirtebiliriz ( Full, Transaction Log  vb.)

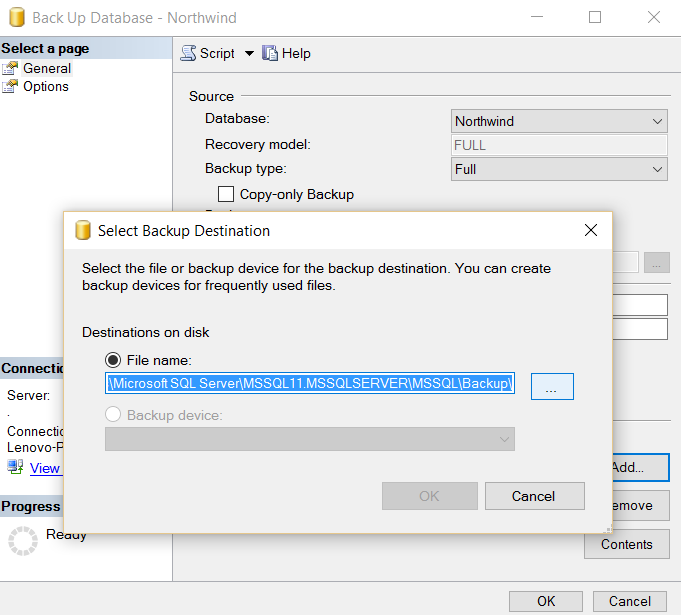
* **Full Backup:** Adından anlaşılacağı üzere Backup ( Yedek ) anı itibariyle Database komple Backup ( Yedek ) içine kopyalanır. Full Backup dosyasi ile başka bir şeye ihtiyaç duymadan Backup ( Yedek ) alındığı tarihe veritabanı Restore ( Geri Yükleme ) işlemi yapılabilir.
* **Differential Backup:** Çok fazla verinin bulunduğu boyutlari yüksek ve sürekli olarak data işlenen Database sürekli olarka Full backup ile Backup ( Yedek ) işlemi yapmak, hem uzun işlem süresi hem de gereksiz disk kullanımlarına neden olur. Böyle durumlarda sadece değişen kısımların yedeğinin alınması Differential Backup seçeneği ile yapabilirsiniz. Ama öncelikle Full Backup ‘ın alınması gerekmektedir.
* **Transaction Log Backup:** Transaction, ilgili Database üzerinde yapılan her işlemin bilgilerinin tutulduğu log dosyasıdır. Yedekleme sadece bu kayıtları kapsar.

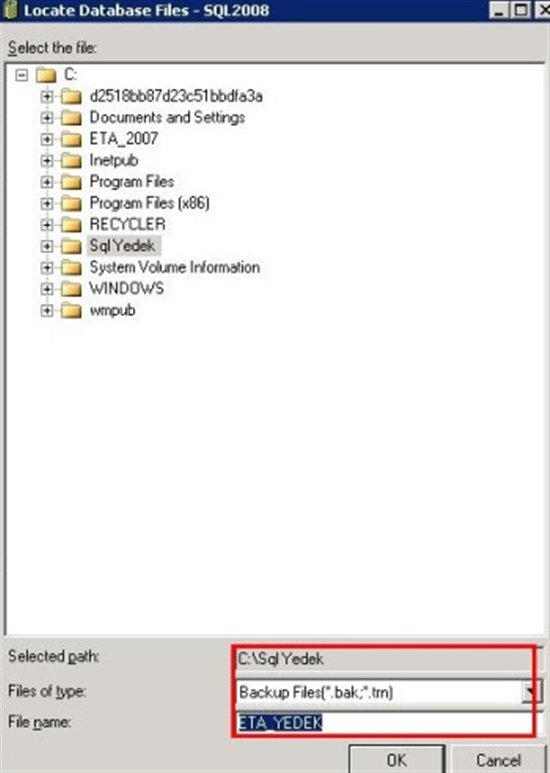
Backup Set : Alınacak yedeğin ismi, açıklaması vb bilgileri belirtebiliriz.

Destination : Alınacak yedeğin sabit disk ya da tape backup donanımında  nereye alınacağı bilgisini görebilir, ekleme ve değişiklik yapabiliriz.

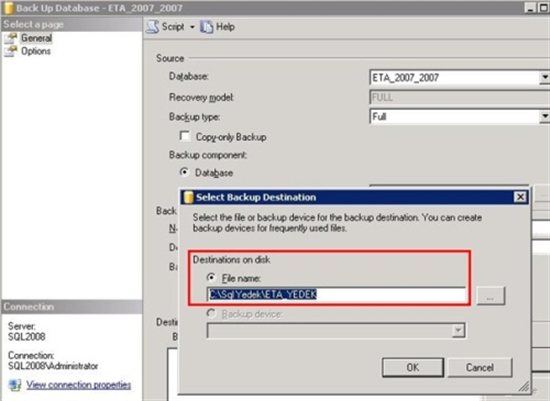


Varsayılan olarak backup işleminden sonra oluşturulacak dosyalar  c:\Program Files \ Microsoft Sql Server  klasörünün altında yer alıyor. Oluşacak backup dosyasının oluşacağı dizin bilgisini değiştirmek için Destination ‘da bulunan Add butonuna tıklıyoruz. Karşımıza Select Backup Destination ekranı geliyor. Burada belirtilen dizin bilgisini değiştireceğimiz için File Name kısmında bulunun  **…**butonuna tıklıyoruz.

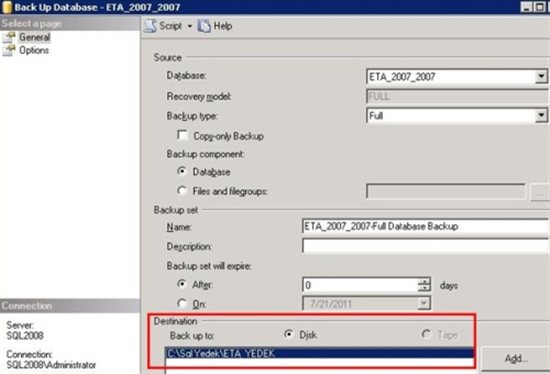


Ekranında  C sürücüsünün altında bulunan “ Sql Yedek “ klasörünü seçerek File Name bilgilerini belirtip bulunduğumuz ekranı OK ile onaylıyoruz.[](http://social.technet.microsoft.com/wiki/cfs-file.ashx/__key/communityserver-wikis-components-files/00-00-00-00-05/2570.004.jpg)

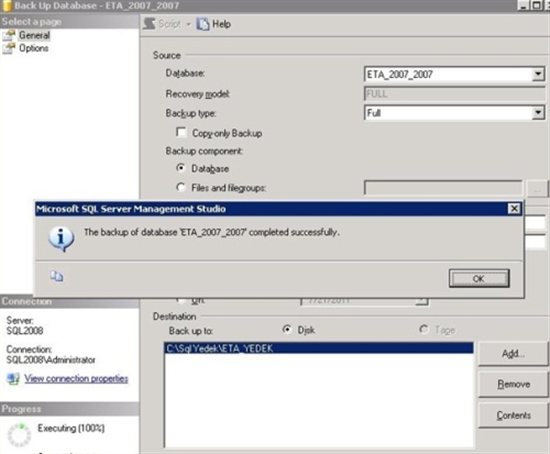
Ve bu işlem ile birlikte tekrar Select Backup Destination ekranına dönüyoruz.  ( İsterseniz doğrudan File Name bölümüne yedek alınacak dizin ve isim bilgilerini yazarak devam edebilirsiniz )

[](http://social.technet.microsoft.com/wiki/cfs-file.ashx/__key/communityserver-wikis-components-files/00-00-00-00-05/6014.005.jpg)

Onaylama işlemi ile birlikte Back Up Database ana ekranına dönüyoruz.

[](http://social.technet.microsoft.com/wiki/cfs-file.ashx/__key/communityserver-wikis-components-files/00-00-00-00-05/7041.006.jpg)

backup işlemini başlatma için Ok ile onaylıyoruz.

[](http://social.technet.microsoft.com/wiki/cfs-file.ashx/__key/communityserver-wikis-components-files/00-00-00-00-05/8688.012.jpg)  
  
Aldığımız yedeği kontrol etmek için “C:\Sql Yedek “  dizin içerisine baktığımızda şekildeki gibi yedeğin sağlıklı bir şekilde alındığını görüyoruz.

**SQL RESTORE**

Restore🡪Veritabanı geri yükleme

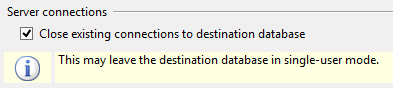
Back-up’ın karşılığı Restore’dur.Back-up’ı geri almak için;

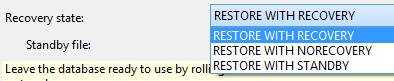
Database 🡪 sağ tık 🡪 Restore 🡪 Database

***General***  Device seçilir ve dosya eklenir 🡪 OK

***Options***  Eğer database sistemde mevcut ise -- Overwrite the existing database seçilir.

Restore yaparken “database şu an kullanılıyor” hatası alıyorsak options kısmında server connection’ı kontrol etmek gerekir.



restore with recovery : ??

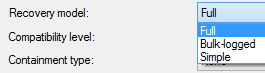
restore with norecovery : back-up alırken kimse erişmesin diye seçilir.

restore with standby : ??

RESTORE DATABASE MarketDB WITH RECOVERY --- Restore’u durdurmak için.

Çalışan bir sistemin restore edilmesi için çok büyük bir sıkıntı olması gerekir. Çok yapılmaz.

database 🡪 sağ tık 🡪 prop. 🡪 options

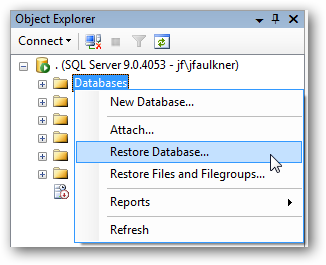
Recovery Model : .ldf’in nasıl kullanılacağını belirler

Full :

Simple :

Bull-logged : pek kullanılmaz

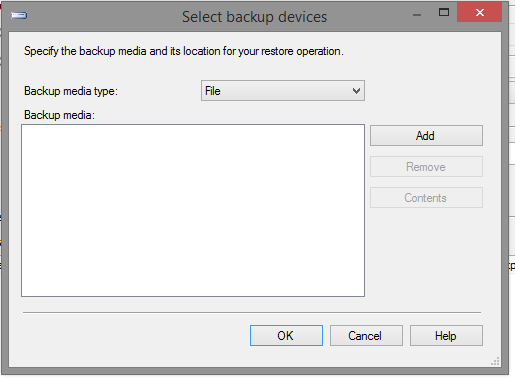
Database klasörüne sağ tıklayın ve Restore Database seçiniz.

[](http://cdn5.howtogeek.com/wp-content/uploads/sg/2010/03/image1.png)

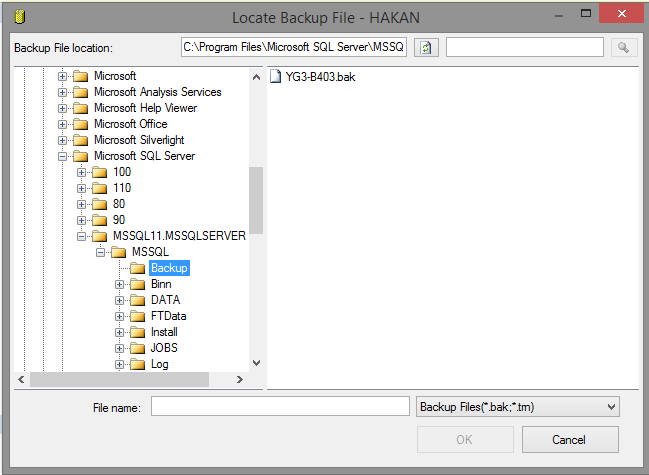
“Source” bölümünden Device seçilip üç noktaya tıklayınız

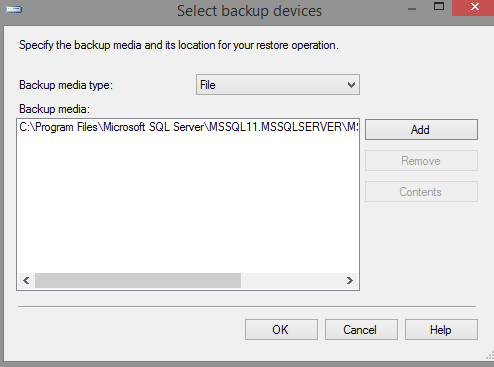


Select backup devices sekmesinden Add tıklayarak dosyalarımızı göreceğiz

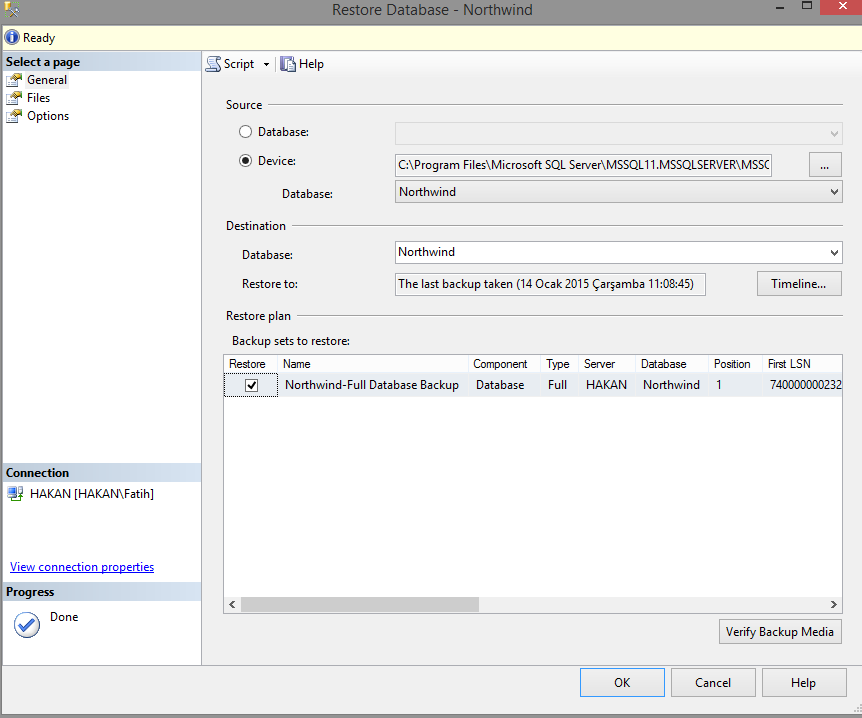


Almak istediğiniz dosyayı seçip ok deyiniz.





Ok deyip devam ediyoruz.

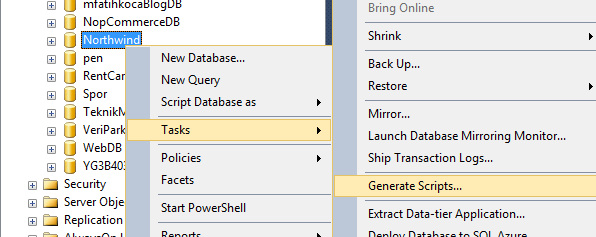


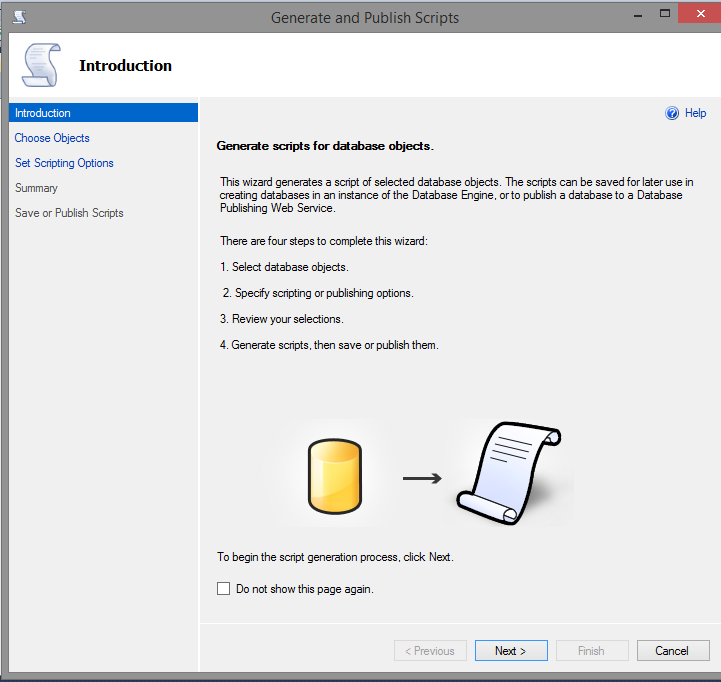
Gelen ekrandada ok dedikten sonra işlem bitecektir.

**Generate Script**

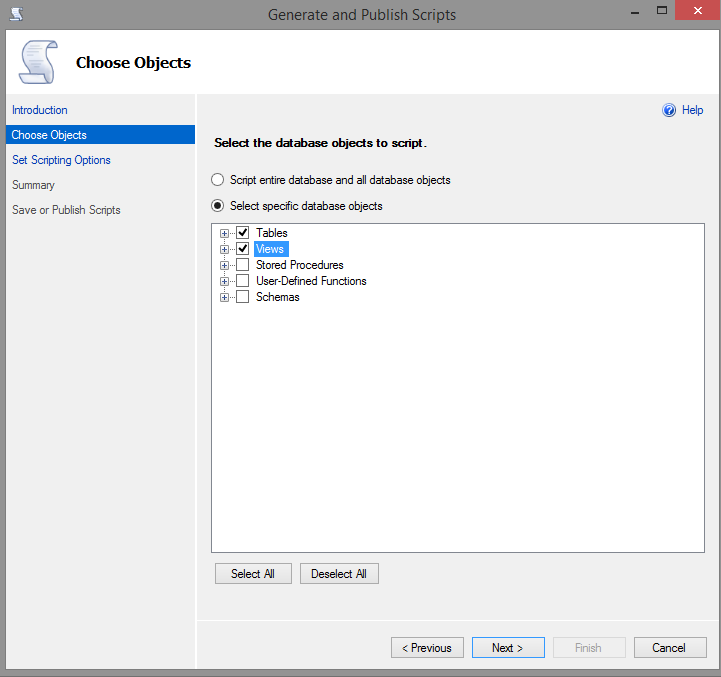
Veritabanını script olarak çıkarmaktan bahsedeceğiz

Veritabanına sağ tıklayıp ilgili yere gelinir.

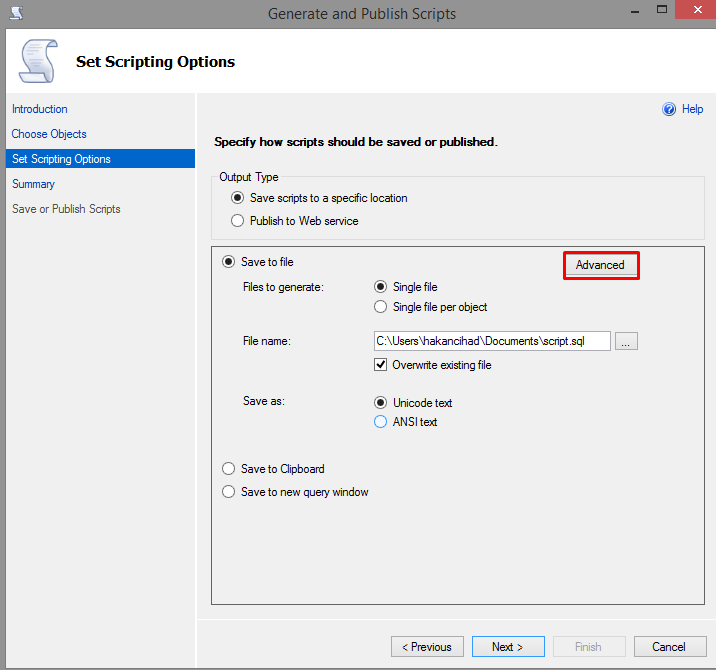




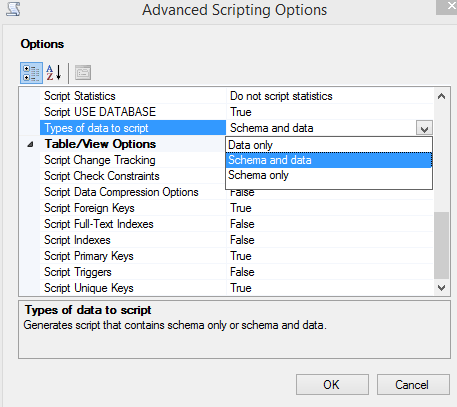
Next deriz



Veritanının istediğimiz bölümlerini alırız.

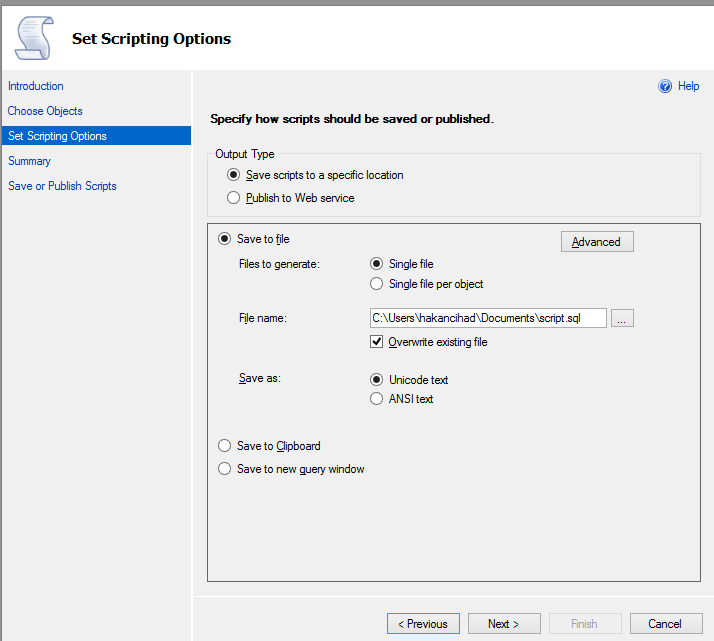


Advanced düğmesine tıklarız

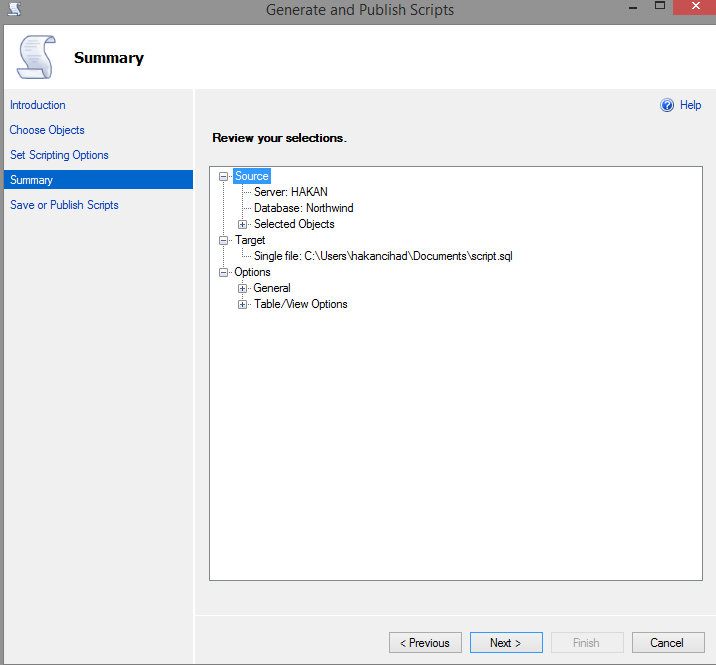


Types of data to script bölümünde Schema and data yı seçersecek aynı zamanda veritabanımızdaki verileride almamızı sağlar.

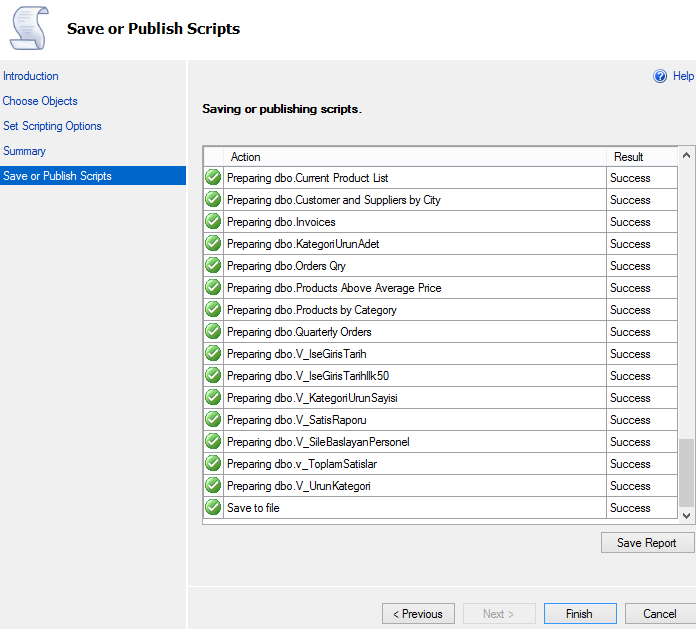
Ok deyip devam ederiz



Ana ekrana gelip next deriz



Tekrardan next deriz



Finish deyip bitiririz.