**بسمه تعالی**



برنامه نویسی بانک اطلاعاتی SQL Server

**نویسنده : محمد حسین فخرآوری**

**جهت گرفتن پشتیبان BACKUP**

|  |
| --- |
| BACKUP DATABASE Data1 TO DISK = 'D:\Data1.bak' |
| BACKUP DATABASE Data1 TO DISK = 'D:\Data1.bak'  WITH -----🡪 Options   * NO\_COMPRESSION * COMPRESSION * DIFFERENTIAL |

**نمایش BACKUP های ایجاد شده**

* SELECT \* FROM msdb.dbo.backupset ORDER BY backup\_set\_id DESC
* SELECT \* FROM msdb.dbo.backupmediafamily ORDER BY physical\_device\_name

**بازیابی اطلاعات پشتیبان گیری LABELONLY, HEADERONLY, FILELISTONLY**

|  |
| --- |
| * RESTORE LABELONLY FROM DISK = 'D:\Data1.bak' * RESTORE HEADERONLY FROM DISK = 'D:\Data1.bak' * RESTORE FILELISTONLY FROM DISK = 'D:\Data1.bak' |

**کاربرد دستور HEADERONLY**

**به شما اجازه می دهد تا اطلاعات تیتر بک آپ را برای همه ی بک آپ های یک دستگاه بک آپ خاص مشاهده کنید. بنابراین در یسیاری موارد هر بک آپ که شما ایجاد می کنید، دارای تنها یک بک آپ ذخیره شده در یک فایل فیزیکی می باشد، بنابراین شما احتمالا تنها یک تیتر می بینید، اما اگر اگر چند بک آپ در یک فایل داشته باشید، اطلاعاتی در مورد هر بک آپ مشاهده خواهید کرد.**

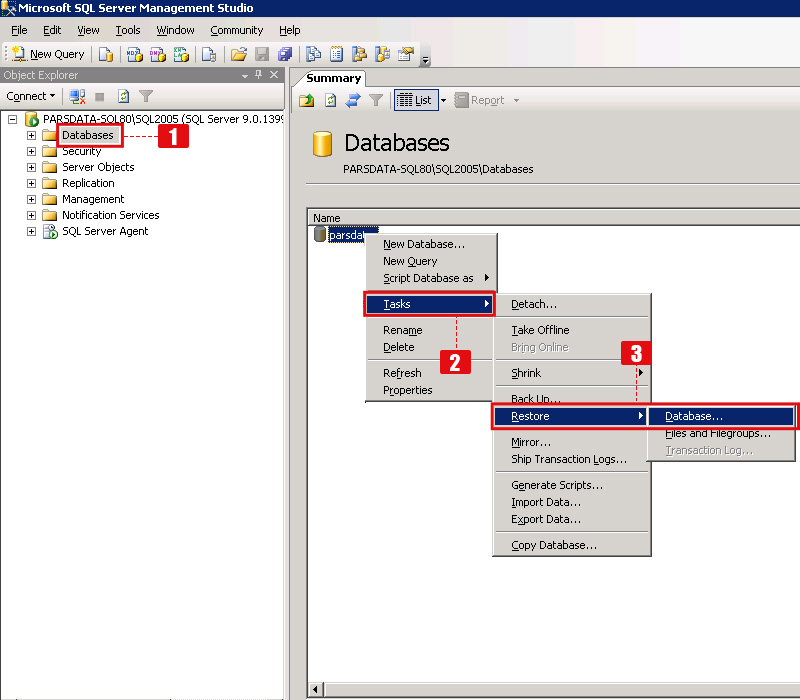
| **Column name** | **Data type** | **Description for SQL Server backup sets** |
| --- | --- | --- |
| BackupName | nvarchar(128) | Backup set name. |
| BackupDescription | nvarchar(255) | Backup set description. |
| BackupType | smallint | Backup type:   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | 1 = Database | 2 = Transaction log | 4 = File | 5 = Differential database | 6 = Differential file | 7 = Partial | 8 = Differential partial | |
| ExpirationDate | datetime | Expiration date for the backup set. |
| Compressed | BYTE(1) | Whether the backup set is compressed using software-based compression:    0 = No  1 = Yes |
| Position | smallint | Position of the backup set in the volume (for use with the FILE = option). |
| DeviceType | tinyint | Number corresponding to the device used for the backup operation.   |  |  |  | | --- | --- | --- | | Disk:  2 = Logical  102 = Physical | Tape:  5 = Logical  105 = Physical | Virtual Device:  7 = Logical  107 = Physical |   Logical device names and device numbers are in sys.backup\_devices; for more information, see[sys.backup\_devices (Transact-SQL)](https://msdn.microsoft.com/en-us/library/ms178018.aspx). |
| UserName | nvarchar(128) | User name that performed the backup operation. |
| ServerName | nvarchar(128) | Name of the server that wrote the backup set. |
| DatabaseName | nvarchar(128) | Name of the database that was backed up. |
| DatabaseVersion | int | Version of the database from which the backup was created. |
| DatabaseCreationDate | datetime | Date and time the database was created. |
| BackupSize | numeric(20,0) | Size of the backup, in bytes. |
| FirstLSN | numeric(25,0) | Log sequence number of the first log record in the backup set. |
| LastLSN | numeric(25,0) | Log sequence number of the next log record after the backup set. |
| CheckpointLSN | numeric(25,0) | Log sequence number of the most recent checkpoint at the time the backup was created. |
| DatabaseBackupLSN | numeric(25,0) | Log sequence number of the most recent full database backup.  DatabaseBackupLSN is the “begin of checkpoint” that is triggered when the backup starts.  This LSN will coincide with FirstLSN if the backup is taken when the database is idle and no replication is configured. |
| BackupStartDate | datetime | Date and time that the backup operation began. |
| BackupFinishDate | datetime | Date and time that the backup operation finished. |
| SortOrder | smallint | Server sort order. This column is valid for database backups only. Provided for backward compatibility. |
| CodePage | smallint | Server code page or character set used by the server. |
| UnicodeLocaleId | int | Server Unicode locale ID configuration option used for Unicode character data sorting. Provided for backward compatibility. |
| UnicodeComparisonStyle | int | Server Unicode comparison style configuration option, which provides additional control over the sorting of Unicode data. Provided for backward compatibility. |
| CompatibilityLevel | tinyint | Compatibility level setting of the database from which the backup was created. |
| SoftwareVendorId | int | Software vendor identification number. For SQL Server, this number is 4608 (or hexadecimal 0x1200). |
| SoftwareVersionMajor | int | Major version number of the server that created the backup set. |
| SoftwareVersionMinor | int | Minor version number of the server that created the backup set. |
| SoftwareVersionBuild | int | Build number of the server that created the backup set. |
| MachineName | nvarchar(128) | Name of the computer that performed the backup operation. |
| Flags | int | Individual flags bit meanings if set to 1:   1 = Log backup contains bulk-logged operations.  2 = Snapshot backup.  4 = Database was read-only when backed up.  8 = Database was in single-user mode when backed up.  16 = Backup contains backup checksums.  32 = Database was damaged when backed up, but the backup operation was requested to continue despite errors.  64 = Tail log backup.  128 = Tail log backup with incomplete metadata.  256 = Tail log backup with NORECOVERY.  Important: We recommend that instead of Flags you use the individual Boolean columns (listed below starting with HasBulkLoggedData and ending with IsCopyOnly). |
| BindingID | uniqueidentifier | Binding ID for the database. This corresponds to sys.database\_recovery\_status\*\*\*\*database\_guid. When a database is restored, a new value is assigned. Also see FamilyGUID (below). |
| RecoveryForkID | uniqueidentifier | ID for the ending recovery fork. This column corresponds to last\_recovery\_fork\_guid in the [backupset](https://msdn.microsoft.com/en-us/library/ms186299.aspx) table.  For data backups, RecoveryForkID equals FirstRecoveryForkID. |
| Collation | nvarchar(128) | Collation used by the database. |
| FamilyGUID | uniqueidentifier | ID of the original database when created. This value stays the same when the database is restored. |
| HasBulkLoggedData | bit | 1 = Log backup containing bulk-logged operations. |
| IsSnapshot | bit | 1 = Snapshot backup. |
| IsReadOnly | bit | 1 = Database was read-only when backed up. |
| IsSingleUser | bit | 1 = Database was single-user when backed up. |
| HasBackupChecksums | bit | 1 = Backup contains backup checksums. |
| IsDamaged | bit | 1 = Database was damaged when backed up, but the backup operation was requested to continue despite errors. |
| BeginsLogChain | bit | 1 = This is the first in a continuous chain of log backups. A log chain begins with the first log backup taken after the database is created or when it is switched from the Simple to the Full or Bulk-Logged Recovery Model. |
| HasIncompleteMetaData | bit | 1 = A tail-log backup with incomplete meta-data.  For information about tail-log backups with incomplete backup metadata, see [Tail-Log Backups (SQL Server)](https://msdn.microsoft.com/en-us/library/ms179314.aspx). |
| IsForceOffline | bit | 1 = Backup taken with NORECOVERY; the database was taken offline by backup. |
| IsCopyOnly | bit | 1 = A copy-only backup.  A copy-only backup does not impact the overall backup and restore procedures for the database. For more information, see [Copy-Only Backups (SQL Server)](https://msdn.microsoft.com/en-us/library/ms191495.aspx). |
| FirstRecoveryForkID | uniqueidentifier | ID for the starting recovery fork. This column corresponds to first\_recovery\_fork\_guid in the [backupset](https://msdn.microsoft.com/en-us/library/ms186299.aspx) table.  For data backups, FirstRecoveryForkID equals RecoveryForkID. |
| ForkPointLSN | numeric(25,0) NULL | If FirstRecoveryForkID is not equal to RecoveryForkID, this is the log sequence number of the fork point. Otherwise, this value is NULL. |
| RecoveryModel | nvarchar(60) | Recovery model for the Database, one of:   |  |  |  | | --- | --- | --- | | FULL | BULK-LOGGED | SIMPLE | |
| DifferentialBaseLSN | numeric(25,0) NULL | For a single-based differential backup, the value equals the FirstLSN of the differential base; changes with LSNs greater than or equal to DifferentialBaseLSN are included in the differential.  For a multi-based differential, the value is NULL, and the base LSN must be determined at the file level. For more information, see [RESTORE FILELISTONLY (Transact-SQL)](https://msdn.microsoft.com/en-us/library/ms173778.aspx).  For non-differential backup types, the value is always NULL. For more information, see [Differential Backups (SQL Server)](https://msdn.microsoft.com/en-us/library/ms175526.aspx). |
| DifferentialBaseGUID | uniqueidentifier | For a single-based differential backup, the value is the unique identifier of the differential base.  For multi-based differentials, the value is NULL, and the differential base must be determined per file.  For non-differential backup types, the value is NULL. |
| BackupTypeDescription | nvarchar(60) | Backup type as string, one of:   * DATABASE * TRANSACTION LOG * FILE OR FILEGROUP * DATABASE DIFFERENTIAL * FILE DIFFERENTIAL PARTIAL * PARTIAL DIFFERENTIAL |
| BackupSetGUID | uniqueidentifier NULL | Unique identification number of the backup set, by which it is identified on the media. |
| CompressedBackupSize | bigint | Byte count of the backup set. For uncompressed backups, this value is the same as BackupSize.  To calculate the compression ratio, use CompressedBackupSize and BackupSize.  During an msdb upgrade, this value is set to match the value of the BackupSize column. |
| containment | tinyint not NULL | Applies to: SQL Server 2012 through SQL Server 2016. Indicates the containment status of the database.   |  |  | | --- | --- | | 0 = database containment is off | 1 = database is in partial containment | |
| KeyAlgorithm | nvarchar(32) | Applies to: SQL Server (SQL Server 2014 (CU1) through current version.  The encryption algorithm used to encrypt the backup. NO\_Encryption indicates that the backup was not encrypted. When the correct value cannot be determined the value should be NULL. |
| EncryptorThumbprint | varbinary(20) | Applies to: SQL Server (SQL Server 2014 (CU1) through current version.  The thumbprint of the encryptor which can be used to find certificate or the asymmetric key in the database. When the backup was not encrypted, this value is NULL. |
| EncryptorType | nvarchar(32) | Applies to: SQL Server (SQL Server 2014 (CU1) through current version.  The type of encryptor used: Certificate or Asymmetric Key. When the backup was not encrypted, this value is NULL. |

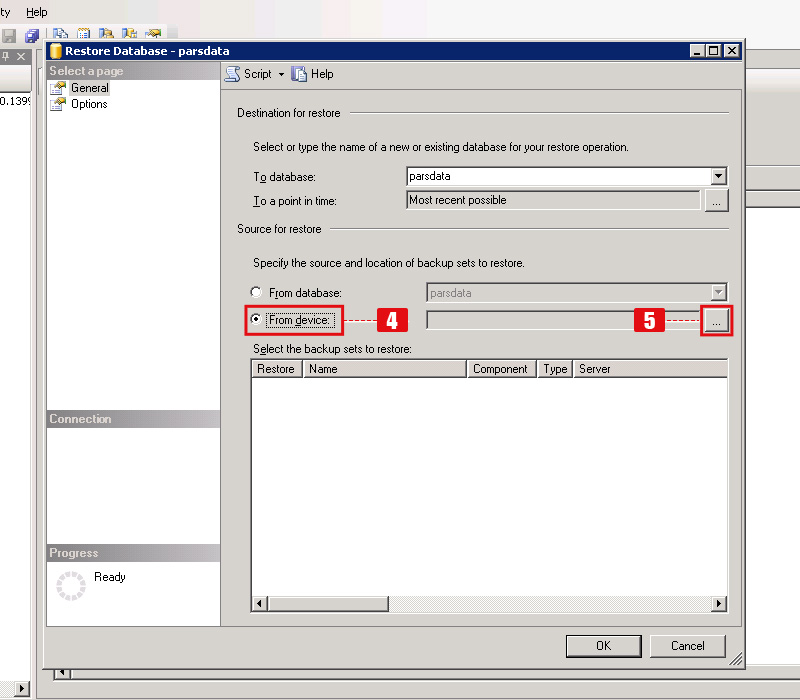
* اگر این فایل بک آپ حاوی چند بک آپ باشد، شما برای هر بک آپ موجود در فایل اطلاعاتی دریافت خواهید کرد. باید خاطر نشان کنیم که بازگردانی  Headeronly به ما می گوید که در موقعیت 1 یک بک آپ کامل و در موقعیت 2  یک بک آپ  Differential داریم.

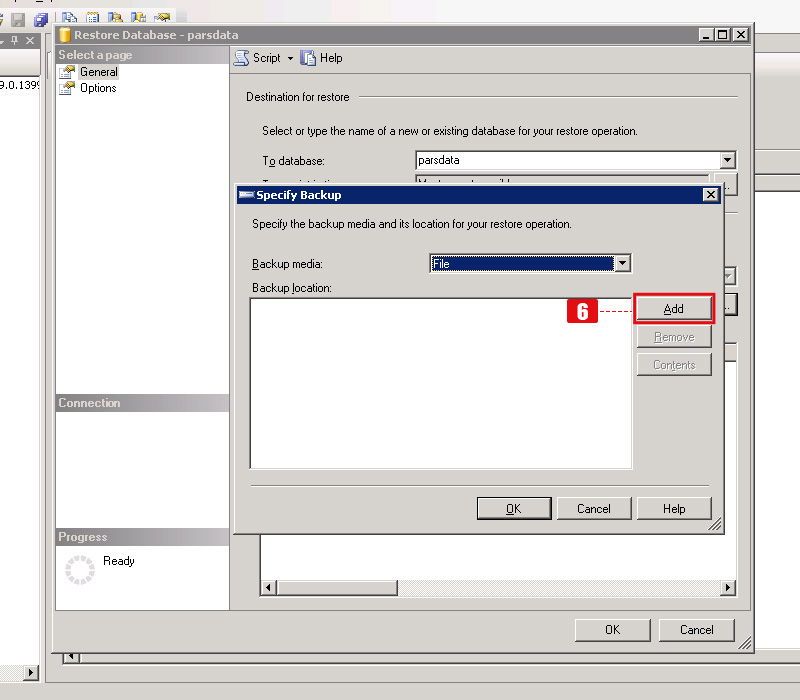
|  |
| --- |
| RESTORE DATABASE AdventureWorks FROM DISK = 'C:\AdventureWorks.BAK' WITH ‎NORECOVERY, FILE = 1‎ |
| RESTORE DATABASE AdventureWorks FROM DISK = 'C:\AdventureWorks.BAK' WITH FILE ‎‎= 2‎ |

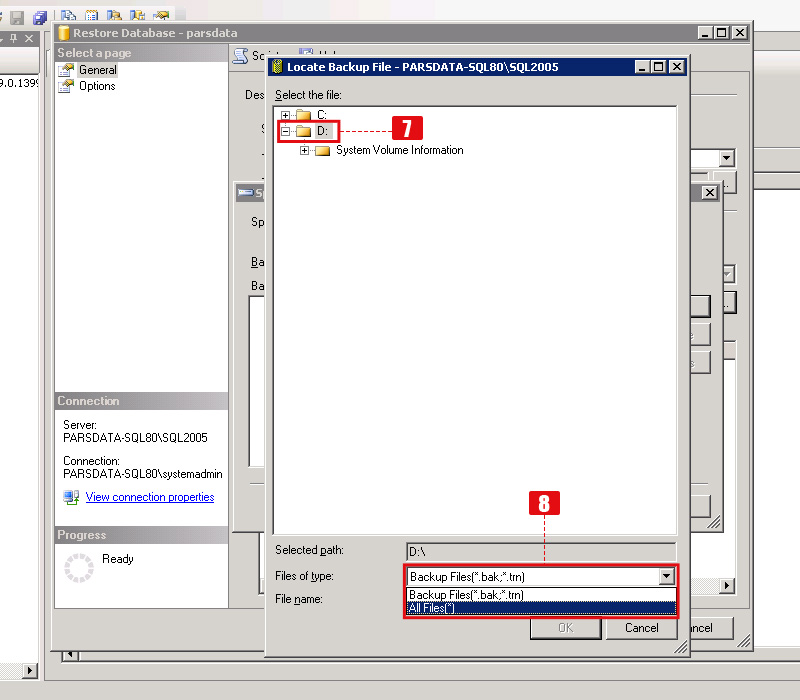
نحوه بازنشانی دیتابیس از طریق Microsoft SQL Server Management Studio

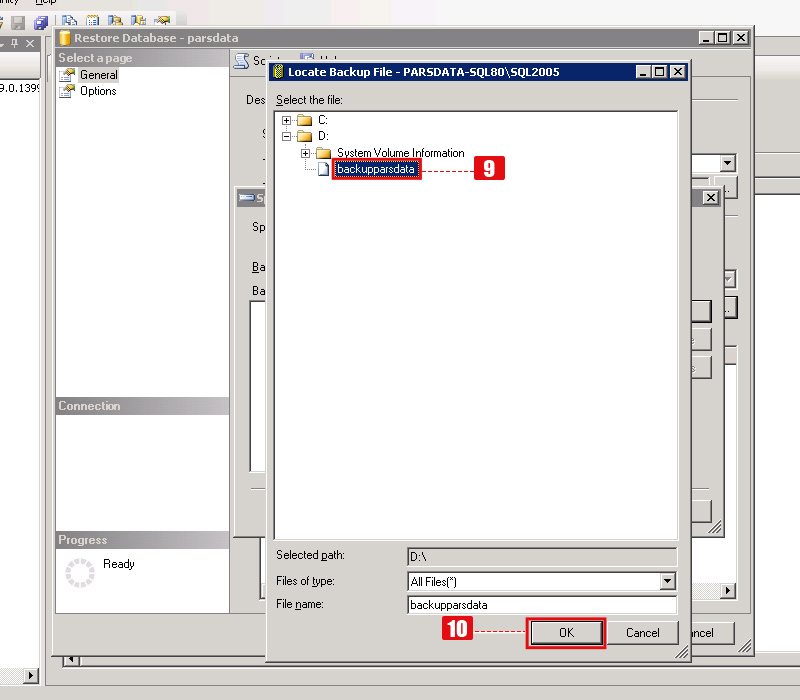
جهت Restore دیتابیس، می توانید از راهنمای تصویری ذیل استفاده نمائید.

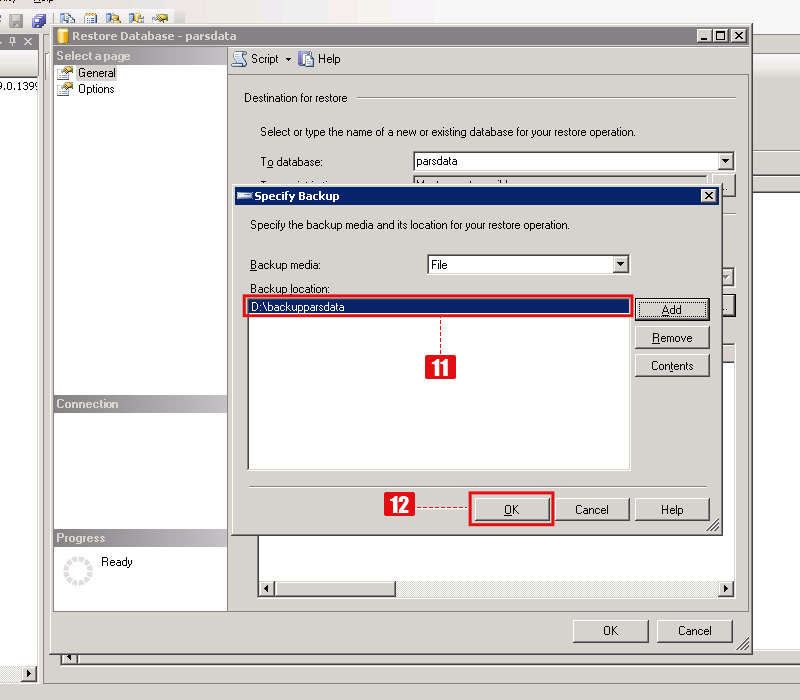


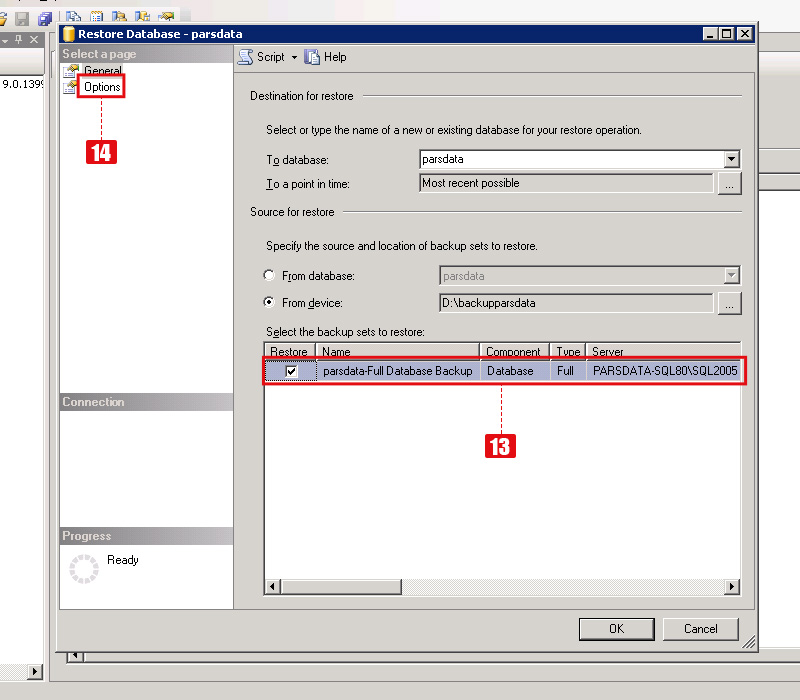


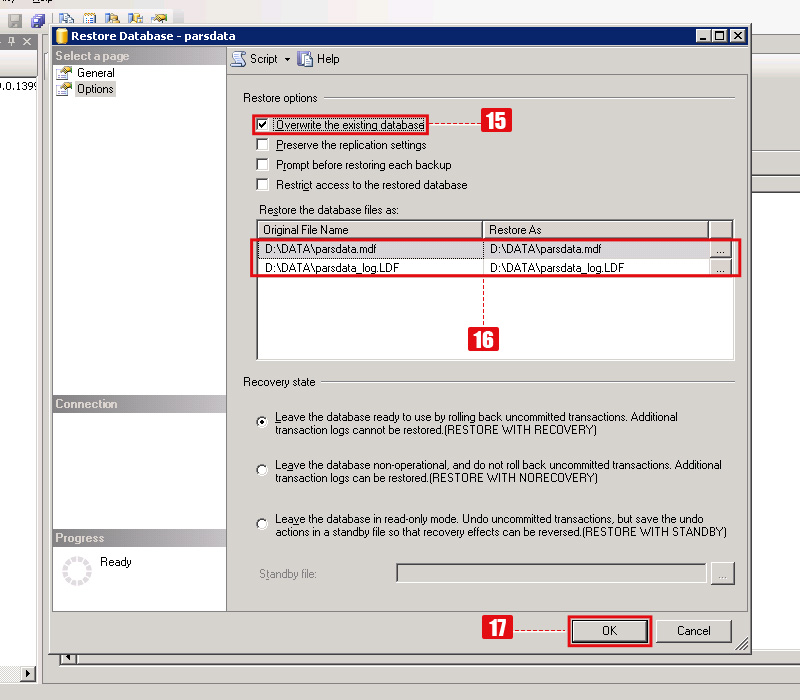


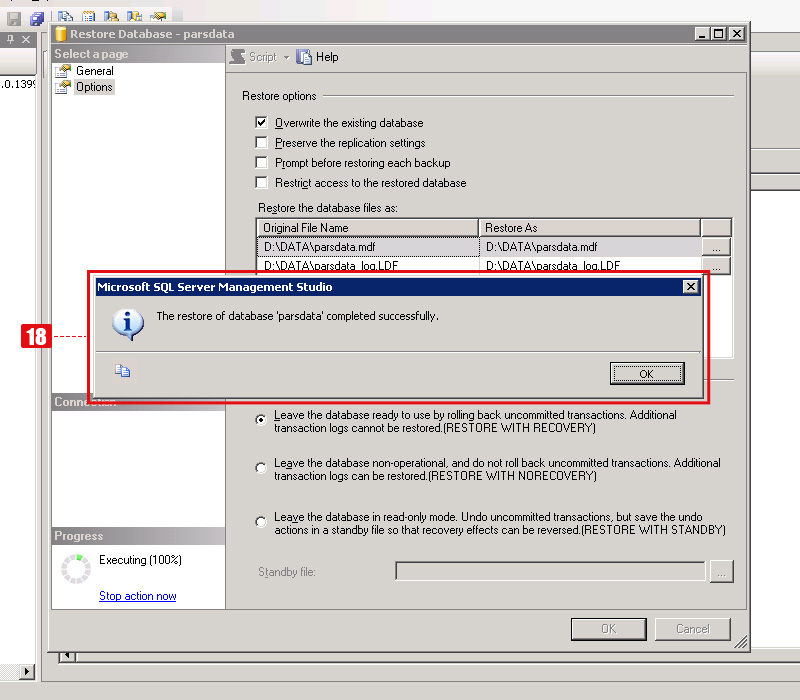












**کاربرد دستور LABELONLY**

به شما اجازه می دهد تا اطلاعات بک آپ مدیا را برای دستگاه بک آپ ببینید. بنابراین اگر یک دستگاه بک آپ مانند یک فایل بک آپ، دارای بک آپ های متعددی است، شما تنها رکورد یکی را دریافت خواهید کرد که به شما اطلاعاتی در مورد مجموعه ی مدیا از قبیل نرم افزاری که برای ایجاد بک آپ استفاده شد، تاریخ ایجاد مدیا و غیره را می دهد.

| Column name | Data type | Description |
| --- | --- | --- |
| MediaName | nvarchar(128) | Name of the media. |
| MediaSetId | uniqueidentifier | Unique identification number of the media set. |
| FamilyCount | int | Number of media families in the media set. |
| FamilySequenceNumber | int | Sequence number of this family. |
| MediaFamilyId | uniqueidentifier | Unique identification number for the media family. |
| MediaSequenceNumber | int | Sequence number of this media in the media family. |
| MediaLabelPresent | tinyint | Whether the media description contains:   1 = Microsoft Tape Format media label  0 = Media description |
| MediaDescription | nvarchar(255) | Media description, in free-form text, or the Tape Format media label. |
| SoftwareName | nvarchar(128) | Name of the backup software that wrote the label. |
| SoftwareVendorId | int | Unique vendor identification number of the software vendor that wrote the backup. |
| MediaDate | datetime | Date and time the label was written. |
| Mirror\_Count | int | Number of mirrors in the set (1-4).  Note: The labels written for different mirrors in a set are identical. |
| IsCompressed | bit | Whether the backup is compressed:  0 = not compressed 1 =compressed |

**کاربرد دستور FILELISTONLY**

به ما اجازه می دهد تا لیستی از فایل هایی که بک آپ گرفته شده اند را مشاهده کنیم. بنابراین برای مثال اگر شما یک بک آپ کامل داشته باشید، تمام فایل های داده (mdf)  و فایل هایlog (ldf)  را مشاهده خواهید کرد.  اگر در یک فایل چند بک آپ وجود داشته باشد و شما "WITH FILE = X"  را مشخص نکنید، تنها اطلاعاتی برای اولین بک آپ در فایل دریافت خواهید کرد.  
برای به دست آوردن تعداد  FILE از RESTORE HEADERONLY  و ستون  "Position" استفاده کنید.

| Column name | Data type | Description |
| --- | --- | --- |
| LogicalName | nvarchar(128) | Logical name of the file. |
| PhysicalName | nvarchar(260) | Physical or operating-system name of the file. |
| Type | char(1) | The type of file, one of:   L = Microsoft SQL Server log file  D = SQL Server data file  F = Full Text Catalog  S = FileStream, FileTable, or In-Memory OLTP container |
| FileGroupName | nvarchar(128) | Name of the filegroup that contains the file. |
| Size | numeric(20,0) | Current size in bytes. |
| MaxSize | numeric(20,0) | Maximum allowed size in bytes. |
| FileID | bigint | File identifier, unique within the database. |
| CreateLSN | numeric(25,0) | Log sequence number at which the file was created. |
| DropLSN | numeric(25,0) NULL | The log sequence number at which the file was dropped. If the file has not been dropped, this value is NULL. |
| UniqueID | uniqueidentifier | Globally unique identifier of the file. |
| ReadOnlyLSN | numeric(25,0) NULL | Log sequence number at which the filegroup containing the file changed from read-write to read-only (the most recent change). |
| ReadWriteLSN | numeric(25,0) NULL | Log sequence number at which the filegroup containing the file changed from read-only to read-write (the most recent change). |
| BackupSizeInBytes | bigint | Size of the backup for this file in bytes. |
| SourceBlockSize | int | Block size of the physical device containing the file in bytes (not the backup device). |
| FileGroupID | int | ID of the filegroup. |
| LogGroupGUID | uniqueidentifier NULL | NULL. |
| DifferentialBaseLSN | numeric(25,0) NULL | For differential backups, changes with log sequence numbers greater than or equal to DifferentialBaseLSN are included in the differential.  For other backup types, the value is NULL. |
| DifferentialBaseGUID | uniqueidentifier | For differential backups, the unique identifier of the differential base.  For other backup types, the value is NULL. |
| IsReadOnly | bit | 1 = The file is read-only. |
| IsPresent | bit | 1 = The file is present in the backup. |
| TDEThumbprint | varbinary(32) | Shows the thumbprint of the Database Encryption Key. The encryptor thumbprint is a SHA-1 hash of the certificate with which the key is encrypted. For information about database encryption, see [Transparent Data Encryption (TDE)](https://msdn.microsoft.com/en-us/library/bb934049.aspx). |
| SnapshotURL | nvarchar(360) | The URL for the Azure snapshot of the database file contained in the FILE\_SNAPSHOT backup. Returns NULL if no FILE\_SNAPSHOT backup. |

**نتیجه**

هنگامی که شما با نسخه پشتیبان مرموز و ناشناخته مواجه شده اید، عصبی نشوید. شما می توانید با LABELONLY، HEADERONLY و FILELISTONLY کنار هم قرار دادن اطلاعات مورد نیاز، بهترین اطلاعات در حال رفتن به خارج از HEADERONLY و FILELISTONLY می آیند و با این شما باید قادر به کشف کردن آنچه که مورد نیاز برای بازگرداندن پایگاه داده باشد.

**بررسی درستی پشتیبان VERIFYONLY**

|  |
| --- |
| RESTORE VERIFYONLY FROM DISK='D:\Test.bak'  **در صورت درستی پیغام** The backup set on file 1 is valid.  RESTORE VERIFYONLY FROM DISK='D:\55.bak'  **در صورت خطا پیغام** VERIFY DATABASE is terminating abnormally. |

**بررسی یک فایل بک آپ روی دیسک برای یک بک آپ خاص**

این فرمان دومین بک آپ را در این فایل بک آپ بررسی خواهد کرد. برای بررسی محتوا در یک فایل بک آپ می توانید از RESTORE HEADERONLY استفاده کنید و از ستون Position برای تعیین تعداد FILE استفاده کنید.

|  |
| --- |
| RESTORE VERIFYONLY FROM DISK='D:\Test.bak' WITH FILE = 2‎ |

* **Detach and Attach Detach) خروج از سرور( Attach) بازگردانی به سرور(**

**Detach**

|  |
| --- |
| EXEC sp\_detach\_db N'Test' |

**Attach**

|  |
| --- |
| EXEC sp\_attach\_db N'Test',  N'C:\Program Files\Microsoft SQL Server\MSSQL10.SQLEXPRESS\MSSQL\DATA\Test.mdf',  N'C:\Program Files\Microsoft SQL Server\MSSQL10.SQLEXPRESS\MSSQL\DATA\Test\_log.ldf' |
| CREATE DATABASE Test ON  (FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL10.SQLEXPRESS\MSSQL\DATA\Test.mdf'),  (FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL10.SQLEXPRESS\MSSQL\DATA\Test\_log.ldf')  FOR ATTACH |

* **گزینه**RESTORE DATABASE  **به ما اجازه می دهد تا یک بک آپ کامل، متفاوت و یا** File Group**را بازگردانی کند.**

**برگشت فایل پشتیبان Bak**

|  |
| --- |
| RESTORE DATABASE Test FROM DISK = 'D:\Test.bak' |

**جایگزینی دیتابیس موجود با فایل Bak قدیمی**

|  |
| --- |
| RESTORE DATABASE Test FROM DISK = 'D:\Test.bak' WITH REPLACE |

**چگونگی بازگردانی یک بک آپ Transaction Log**

در هنگام بازگردانی یک  transaction log نیاز به دسترسی مجزا به  دیتابیس خواهید داشت که به این معناست که اتصالات هیچ کاربر دیگری نمی تواند از دیتابیس استفاده کند. اگر دیتابیس در حالت بازگردانی باشد،  انتشار اتفاق نمی افتد، زیرا هیچ کس نمی تواند از دیتابیس استفاده کند.

|  |
| --- |
| **پشتيبان‌گيري در حالت دستی (Manual)**  با كلیك روی SQL Server Management Studio و وارد كردن نام سرور و اطلاعات كاربری وارد محیط SQL می شویم. |
| Connect to Server |
| از لیست پوشه های قابل مشاهده روی علامت + Databases كلیك كرده و سپس روی پایگاه داده مورد نظرمان ( در اینجا پوشه dbrates ) كلیك راست كرده و مطابق شكل ابتدا روی گزینه Task وسپس گزینه Backup می رویم. |
| Object Explorer |
| با كلیك روی گزینه Backup پنجره مربوط به آن باز می شود. |
| Backup Database |
| مطابق شكل بالا در قسمت Select a Page دو گزینه General و Options برای پیكره بندی كلی فرآیند پشتیبانی وجود دارد. با انتخاب گزینه General مطابق شكل بالا، امكان انتخاب دیتابیس برای پشتیبان گیری، نوع پشتیبان، نام گذاری فایل و تعریف انقضا و مقصد ذخیره فایل وجود دارد که در ادامه بطور كامل توضیح داده خواهد شد. |
| Source |
| در گزینه Database پایگاه داده مورد نظر ( در اینجا dbrates ) را انتخاب می كنیم.  در گزینه Recovery Model حالت Simple را انتخاب می كنیم.  در گزینه Backup Type بسته به نوع پشتیبان، در حالت Full یا Differential قرار می‌دهیم. حالت Full به منزله پشتیبان گیری كامل از دیتابیس، و حالت Differential به منزله پشتیبان گیری از تغییرات و داده‌های اضافه شده به دیتابیس موجود است.  در گزینه Backup Component گزینه Database را انتخاب می كنیم. |
| مطابق شكل در زیر عبارت Backup set موارد زیر قابل انجام است. |
| Backup Set |
| در گزینه Name امكان درج نامی برای این پشتیبان است كه بطور پیش فرض نامی برای آن انتخاب شده است ضمن اینكه امكان ویرایش آن وجود دارد.  در گزینه Backup Set will expire (برای تعیین زمان انقضای دیتابیس پشتیبان شده ) دو انتخاب وجود دارد. تعیین تعداد روز بعد از پشتیبان گیری با انتخاب گزینه After و درج روز (عدد صفر، مقدار پیش فرض و به معنی عدم انقضای فایل پشتیبان است) و یا تعیین تاریخ خاص براساس سال میلادی برای انقضای فایل پشتیبان مورد نظر با انتخاب گزینه On .  در بخش Description اطلاعات فایل پشتیبان با توجه به توضیحات زیر ثبت خواهد شد. |
| Destination |
| در گزینه Back up to امكان مشخص كردن محل ذخیره سازی با توجه به انواع آن ( حالت های Disk یا Tape ) وجود دارد. بطور پیش فرض فایل پشتیبان با نام پیش فرض در مسیر نصب SQL در پوشه Backup و بصورت تك فایل قرار خواهد گرفت. امكان تغییر این مسیر وجود دارد. می توان فایل پشتیبان را در قالب چند فایل جدا از هم ولی بهم پیوسته (در هنگام بازگردان نیاز به حضور تمامی فایل ها است) ایجاد كرد. امكان حذف مسیر پیش فرض فایل پشتیبان (با زدن دکمه Remove ) و سپس اضافه كردن مسیر جدید با زدن كلید Add و تعیین محل ذخیره فایل پشتیبان مطابق شكل زیر، با انتخاب مسیر و نوشتن نام پشتیبان در بخش File name وجود دارد. |
| Select the File |
| مطابق شكل زیر در قسمت Select a Page با انتخاب گزینه Options امكانات پیشرفته‌ی دیگری فراهم می شود. |
| Options |
| با انتخاب گزینه Options ، امكان انتخاب تعیین حالت قرارگیری فایل های متعدد پشتیبان و انتخاب خطایابی در هنگام پروسه پشتیبان گیری وجود دارد که در ادامه بطور كامل توضیح داده خواهد شد.  مطابق شكل بالا در عبارت Overwrite media گزینه ای برای الحاق Append نسخه های پشتیبان در یک فایل وجود دارد، با انتخاب این گزینه اگر فایل انتخابی شما برای پشتیبان گیری، پیش از این وجئود داشته باشد، پشتیبان جدید به همان فایل افزوده خواهد شد. ( این ویژگی در مواقعی كه بیش از یك پشتیبان در طول روز از دیتابیس تهیه می شود كاربرد دارد.) این گزینه بطور پیش فرض فعال است. البته می توان با انتخاب گزینه Overwrite به جای الحاق به فایل موجود، آنها را حذف و پشتیبان جدید را جایگزین قبلی نمود. |
| Overwrite Media |
| با فعال كردن گزینه Check Media set ، وجود نسخه پشتیبان قبلی و زمان انقضای آن بررسی خواهد شد.  گزینه Back up to a new media set, and erase all existing backup sets زمانی انتخاب می شود كه نیازی به نسخه‌های پشتیبان قبلی نداشته باشید و می‌خواهید تمامی آن ها حذف شوند  مطابق شكل در زیر عبارت Reliability امكان اطمینان از صحت پشتیبان گیری بدون اشكال و خطا وجود دارد. |
| Reliability |
| با انتخاب گزینه Verify backup when finished پس از اتمام روند پشتیبان گیری، صحت عملیات و صحت ذخیره شدن نسخه پشتیبان در فایل (مقصد) مورد نظر، بررسی خواهد شد.  انتخاب گزینه Perform checksum before writing to media قابلیت بررسی اعتبار بخش های مختلف اطلاعات را به نسخه پشتیبان می افزاید. این عمل عملیات پشتیبان گیری را سنگین خواهد کرد اما قابلیت نادیده گرفتن خطا در عملیات یا باز یابی اطلاعات را به دست خواهید آورد. با انتخاب گزینه Continue on error در صورت وجود خطا در عملیات، می توان با نادیده گرفتن خطا، به ذخیره سازی دیگر اطلاعات پرداخت.  با فشردن كلید OK عملیات تهیه پشتیبان شروع می شود. |
| Done Dialogue |

[**تفاوت Full, Differential, and Incremental Backup**](http://itexperience.blogfa.com/post/96/تفاوت-Full,-Differential,-and-Incremental-Backup)

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **تعریف** | **مزایا** | **معایب** |
| **Full Backup** | بک آپ کامل از هر چیزی که بخواهید مثلا بک آپ کامل از یک دیتابیس درSQL شامل تمام محتوای دیتابیس در بک آپ است. | بازگردانی سریع در صورتی که آخرین بک آپ مورد نظر در دسترس باشد. | پروسه بک آپ گیری طولانی است و در صورتی که حجم اطلاعات شما بالا باشد نیاز به یک دیسک حجیم برای نگهداری بک آپ ها دارد. |
| **Differential Backup** | مکانیزم این نوع بک آپ بر اساس آخرین فول بک آپ گرفته شده است ، یعنی هر تغییری که بعد از آخرین فول بک آپ صورت می گیرد را ذخیره می کند.  اگر 2 بک آپ  **Differential**  گرفته بشه ، آخرین بک آپ مدنظر ماست برای برگرداندن، نیاز به آخرین فول بک آپ و آخرین diff دارید. | خیلی سریعتر از Full بک آپ می گیرد.  نسبت به incremental برای بازگردانی سرعت بیشتری دارد.  نسبت به فول بک آپ به حجم نگهداری کمتری نیاز دارد. | بازگردانی آن از فول بک آپ آرامتر است.  گرفتن بک آپ در این سناریو از incremental سرعت کمتری دارد. |
| **Incremental Backup** | در این سناریو ، بک آپ از آخرین تغییرات تا وقتی که یکی از 3 بک آپ زیر انجام شده باشد گرفته میشود  (full, differential or incremental) | این متد بهترین سرعت را برای بک آپ گیری به همراه دارد و همچنین کمترین حجم مورد نیاز برای نگهداری | نسبت به دو مورد بالا بازگردانی با سرعت کمتری انجام می شود ، به خاطر اینکه برای بازگردانی شما بجای 1 فایل یا 2 فایل باید چندین فایل را که شامل تغییرات پی در پی دیتابیس بوده برگردانی کنید. |

گرفتن بکاپ با Generate Scripts

