

SPRAWOZDANIE Z ZADANIA 2

Polecenie:

backup wszystkich baz przez procedurę

- Z2
 - ** Napisac 2 procedury
 - ** bk_db - backup pojedynczej bazy (2 par -> nazwa bazy i katalog)
 - ** bk_all_db - backup wszystkich baz (nazwa katalogu)
 - ** do plików na wyznaczonym katalogu
 - ** bk_all_db musi korzystać z bk_db i przekazywać jej nazwę bazy i nazwę katalogu
 - ** nazwa kazdego pliku to nazwabazy PODKRESLENIE YYYYYMMDDHHMM
 - ** Zaplanować uruchamianie procedury backupy wszystkich baz poprzez SQL Agent na co dzien
 - ** Zdokumentować i udowodnić, że JOB zadziałał i pliki powstały
 - */

Kod programu z komentarzami:

```
USE pwx_db
GO

-- procedura backupu pojedynczej bazy
CREATE PROCEDURE bk_db

-- argumenty (nazwa bazy, ścieżka do katalogu)
@db nvarchar(100)
,@path nvarchar(200)

AS
BEGIN

-- zmienna fname czyli nazwa pliku w którym zapisze sie nasz backup
declare @fname nvarchar(1000)
SET @path = LTRIM(RTRIM(@path)) -- pomijamy spacje z obu stron

IF @path NOT LIKE N'%\' -- nie ma \ na końcu
    SET @path = @path + N'\ '

SET @fname = REPLACE(REPLACE(CONVERT(nchar(19), GETDATE()), 126), N':', N'_'), '-', '_')
SET @fname = @path + RTRIM(@db) + @fname + N'.bak'

DECLARE @sql nvarchar(1000)

SET @sql = 'backup database ' + @db + ' to DISK= N''' + @fname + ''''

EXEC sp_sqlexec @sql

END
```

```

-- TEST PROCEDURY bk_db
exec bk_db N'PWX_DB', N'C:\backup'

USE pwx_db
GO

-- procedura do backupu wszystkich baz

CREATE PROCEDURE bk_all_db
-- argument ( ścieżka do katalogu)
@path nvarchar(200)
AS
BEGIN

-- tworzymy tabelę w której będziemy przechowywać nazwy naszych baz
create TABLE #databases_name( name sysname)

insert into #databases_name(name)

SELECT name FROM sys.databases WHERE database_id > 4

DECLARE @cnt INT = 0;

-- tworzymy zmienną w której przechowamy liczbę, która określa nam ilość baz

```

```

declare @total_cnt int
set @total_cnt = (SELECT COUNT(*) FROM #databases_name)

declare @name nvarchar(100)

-- petla po wszystkich bazach
WHILE @cnt < @total_cnt
BEGIN

-- bierzemy nazwę pierwszej bazy
set @name = (SELECT TOP 1* FROM #databases_name)
-- usuwamy pobraną nazwę bazy z tabeli #databases_name
DELETE TOP (1) FROM #databases_name
-- wywołujemy dla pobranej bazy procedure jej backupu (przekazujemy nazwę tej bazy i ścieżkę)
exec bk_db @name, @path
SET @cnt = @cnt +1;

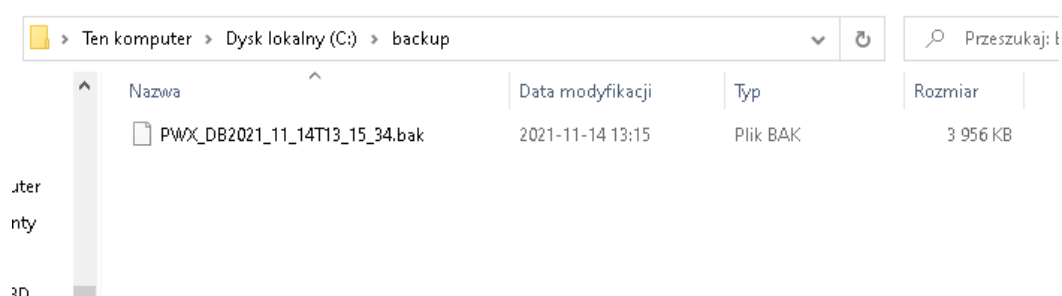
END
END

```

Działanie programu:

- **Dowód poprawnego działania pierwszej procedury (backup wybranej pojedynczej procedury do wskazanego katalogu):**

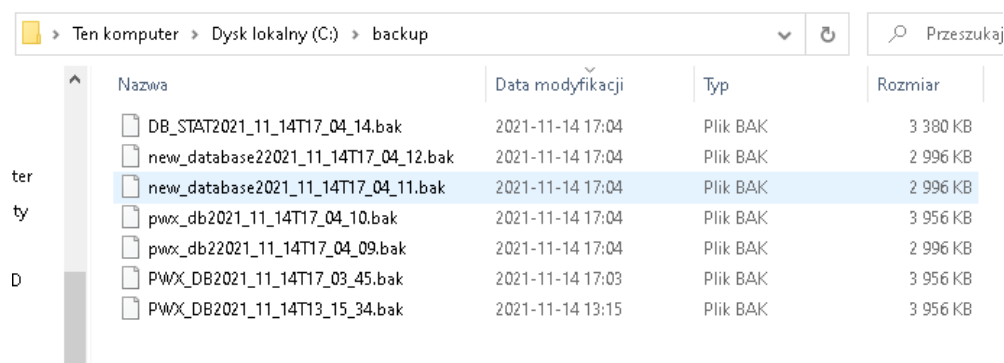
```
exec bk_db N'PWX_DB', N'C:\backup'
```



Ten komputer > Dysk lokalny (C:) > backup					Przeszukaj: t
Nazwa	Data modyfikacji	Typ	Rozmiar		
PWX_DB2021_11_14T13_15_34.bak	2021-11-14 13:15	Plik BAK	3 956 KB		

- **Dowód poprawnego działania drugiej procedury (backup wszystkich procedur (oprócz systemowych) do wskazanego katalogu):**

```
exec bk_all_db N'C:\backup'
```



Ten komputer > Dysk lokalny (C:) > backup					Przeszukaj
Nazwa	Data modyfikacji	Typ	Rozmiar		
DB_STAT2021_11_14T17_04_14.bak	2021-11-14 17:04	Plik BAK	3 380 KB		
new_database2021_11_14T17_04_12.bak	2021-11-14 17:04	Plik BAK	2 996 KB		
new_database2021_11_14T17_04_11.bak	2021-11-14 17:04	Plik BAK	2 996 KB		
pwx_db2021_11_14T17_04_10.bak	2021-11-14 17:04	Plik BAK	3 956 KB		
pwx_db2021_11_14T17_04_09.bak	2021-11-14 17:04	Plik BAK	2 996 KB		
PWX_DB2021_11_14T17_03_45.bak	2021-11-14 17:03	Plik BAK	3 956 KB		
PWX_DB2021_11_14T13_15_34.bak	2021-11-14 13:15	Plik BAK	3 956 KB		

- Wynik poprawnego wykonania drugiej procedury – Messages w MSSQL Server Managment Studio:

```
Messages

(5 rows affected)

(1 row affected)
Processed 360 pages for database 'pwx_db2', file 'pwx_db2' on file 1.
Processed 2 pages for database 'pwx_db2', file 'pwx_db2_log' on file 1.
BACKUP DATABASE successfully processed 362 pages in 1.005 seconds (2.810 MB/sec).

(1 row affected)
Processed 480 pages for database 'pwx_db', file 'pwx_db' on file 1.
Processed 2 pages for database 'pwx_db', file 'pwx_db_log' on file 1.
BACKUP DATABASE successfully processed 482 pages in 0.620 seconds (6.067 MB/sec).

(1 row affected)
Processed 360 pages for database 'new_database', file 'new_database' on file 1.
Processed 2 pages for database 'new_database', file 'new_database_log' on file 1.
BACKUP DATABASE successfully processed 362 pages in 0.981 seconds (2.878 MB/sec).

(1 row affected)
Processed 360 pages for database 'new_database2', file 'new_database2' on file 1.
Processed 2 pages for database 'new_database2', file 'new_database2_log' on file 1.
BACKUP DATABASE successfully processed 362 pages in 1.755 seconds (1.609 MB/sec).

(1 row affected)
Processed 408 pages for database 'DB_STAT', file 'DB_STAT' on file 1.
Processed 2 pages for database 'DB_STAT', file 'DB_STAT_log' on file 1.
BACKUP DATABASE successfully processed 410 pages in 1.254 seconds (2.551 MB/sec).

Completion time: 2021-11-14T17:38:45.8931090+01:00
|
```

- Uruchomienie zaplanowania back-upu wszystkich baz codziennie za pomocą JOB w SQL Server Agent:

New Job

Select a page

- General
- Steps
- Schedules
- Alerts
- Notifications
- Targets

Script ? Help

Name: backup_all_db

Owner: Asus-Komputer\Asus

Category: [Uncategorized (Local)]

Description:

☒ Enabled

Connection

Server: ASUS-KOMPUTER

Connection: Asus-Komputer\Asus

[View connection properties](#)

- Zmieniam Database na używaną przeze mnie pwx_db
- Ustawiam komendę exec bk_all_db (wywołanie procedury, która robi backup wszystkich baz)

New Job Step

Select a page

- General
- Advanced

Script Help

Step name: backup_all_db

Type: Transact-SQL script (T-SQL)

Run as:

Database: pwx_db

Command: exec bk_all_db

Open... Select All Copy Paste Parse

Connection

Server: ASUS-KOMPUTER

Connection: Asus-Komputer\Asus

[View connection properties](#)

Progress

Ready

Previous Next

- Ustawiam schedule, by zadanie (backup wszystkich baz) było wykonywane każdego dnia o 17:30:

- Dowód, że Job zadziałał i o 17:30 został wykonany zaplanowany backup

Nazwa	Data modyfikacji	Typ	Rozmiar
DB_STAT2021_11_14T17_30_07.bak	2021-11-14 17:30	Plik BAK	3 380 KB
new_database22021_11_14T17_30_05.bak	2021-11-14 17:30	Plik BAK	2 996 KB
new_database2021_11_14T17_30_03.bak	2021-11-14 17:30	Plik BAK	2 996 KB
pwxx_db2021_11_14T17_30_01.bak	2021-11-14 17:30	Plik BAK	3 956 KB
pwxx_db2021_11_14T17_30_00.bak	2021-11-14 17:30	Plik BAK	2 996 KB

- Dowód wykonania job activity – widzimy, że ostatnie wywołanie przebiegło pomyślnie, widzimy datę ostatniego wykonania i datę następnego (kolejny dzień ta sama godzina), widzimy, że czynność jest cykliczna (Scheduled == yes)

Name	Enabled	Status	Last Run O...	Last Run	Next Run	Category	Runnable	Scheduled	Category ID
backup_all_db	yes	Idle	Succeeded	2021-11-14 17:30:00	2021-11-15 17:30:00	[Uncategor...	yes	yes	0

Kod programu (wklejony, gdyby była konieczność uruchomienia w programie):

```
USE pwx_db
```

```
GO
```

```
-- procedura backupu pojedynczej bazy
```

```
CREATE PROCEDURE bk_db
```

```
-- argumenty (nazwa bazy, ścieżka do katalogu)
```

```
@db nvarchar(100)
```

```
,@path nvarchar(200)
```

```
AS
```

```
BEGIN
```

```
-- zmienna fname czyli nazwa pliku w którym zapisze sie nasz backup
```

```
declare @fname nvarchar(1000)
```

```
SET @path = LTRIM(RTRIM(@path)) -- pomijamy spacje z obu stron
```

```
IF @path NOT LIKE N'%\' -- nie ma \ na końcu
```

```
SET @path = @path + N'\'
```

```
SET @fname = REPLACE(REPLACE(CONVERT(nchar(19), GETDATE(), 126), N':', N'_'), '-', '_')
```

```
SET @fname = @path + RTRIM(@db) + @fname + N'.bak'
```

```
DECLARE @sql nvarchar(1000)
```

```
SET @sql = 'backup database ' + @db + ' to DISK= N''' + @fname + ''''
```



```
EXEC sp_sqlexec @sql
```

```
END
```

```
GO
```

```
-- TEST PROCEDURE bk_db
```

```
exec bk_db N'PWX_DB', N'C:\backup'
```

```
USE pwx_db
```

```
GO
```

```
-- procedura do backupu wszystkich baz
```

```
CREATE PROCEDURE bk_all_db
```

```
-- argument ( ścieżka do katalogu)
```

```
@path nvarchar(200)
```

```
AS
```

```
BEGIN
```

```
-- tworzymy table w ktorej bedziemy przechowywac nazwy naszych baz
```

```
create TABLE #databases_name( name sysname)
```

```
insert into #databases_name(name)
```

```
SELECT name FROM sys.databases WHERE database_id > 4
```

```
DECLARE @cnt INT = 0;
```

```
-- tworzymy zmienną, w której przechowamy liczbę, która określa nam ilość baz

declare @total_cnt int

set @total_cnt = (SELECT COUNT(*) FROM #databases_name)

declare @name nvarchar(100)

-- petla po wszystkich bazach
WHILE @cnt < @total_cnt
BEGIN

-- bierzemy nazwę pierwszej bazy
set @name = (SELECT TOP 1* FROM #databases_name)
-- usuwamy pobraną nazwę bazy z tabeli #databases_name
DELETE TOP (1) FROM #databases_name
-- wywołujemy dla pobranej bazy procedurę jej backupu (przekazujemy nazwę tej bazy i ścieżkę)
exec bk_db @name, @path

SET @cnt = @cnt +1;

END

END

-- test procedury backup wszystkich baz
exec bk_all_db N'C:\backup'
```