



Syllabus

Databases

Mileto Di Marco
Kristien Roels
Bart Soete

Inhoudsopgave

H9 - Stored procedures	3
1 Inleiding.....	3
2 De basis van stored procedures	3
3 Data doorgeven aan een stored procedure.....	5
4 Data teruggeven vanuit een stored procedure.....	6

H9 - Stored procedures

1 Inleiding

Een stored procedure is een groep van één of meerdere Transact-SQL-statements die opgeslagen is in de databank op de server. Stored procedures zijn gelijkaardig aan procedures of methodes in andere programmeertalen.

Een goed geschreven stored procedure is gebaseerd op een goed geschreven batch en goed geschreven SQL query's. Deze zaken kwamen aan bod in eerdere hoofdstukken. In dit hoofdstuk wordt aangegeven hoe een batch verpakt kan worden als een stored procedure.

Een stored procedure kan naast het afleveren van data (SELECT) ook wijzigingen doorvoeren aan de data (INSERT, UPDATE en DELETE) en aan de databankstructuur (CREATE, ALTER en DROP).

Stored procedures worden veel gebruikt omdat ze een aantal voordelen bieden:

- Minder client/server netwerkverkeer. Inderdaad, alleen de procedure-oproep, en niet de individuele codelijnen, worden over het netwerk gestuurd.
- Sterkere security. Daar enkel de oproep over het netwerk gestuurd wordt, zijn de tabel- en objectnamen en de embedded SQL-statements niet zichtbaar.
- Het gebruik van procedure-parameters helpt dan weer tegen SQL-injection.
- Herbruiken van code. De code voor herhaalde databankoperaties is de perfecte kandidaat voor encapsulatie in een stored procedure. Dit elimineert het herschrijven van dezelfde code, vermindert code inconsistentie en verbergt complexiteit.
- Eenvoudiger onderhoud. Als de client-applicaties stored procedures oproepen, dan moeten enkel de stored procedures aangepast worden als er wijzigingen in de onderliggende databank zijn.
- Betere performantie. Een stored procedure wordt gecompileerd als ze de eerste keer uitgevoerd wordt en er wordt een execution plan bewaard in de cache van de Database Engine dat zal gebruikt worden bij volgende oproepen.

Relevante Help-pagina's:

<https://docs.microsoft.com/nl-nl/sql/relational-databases/stored-procedures/stored-procedures-database-engine>
en

<https://docs.microsoft.com/en-us/sql/t-sql/statements/create-procedure-transact-sql>

De voorbeelddatabank die in dit hoofdstuk gebruikt wordt is de GRADSchool-databank. Vergeet dus niet om van de GRADSchool-databank de actieve databank te maken:

```
USE GRADSchool
```

2 De basis van stored procedures

Het aanmaken en wijzigen van een stored procedure:

```
CREATE [OR ALTER] PROCEDURE procedurenaam
```

```
AS
```

```
SQL-statement [...]
```

Merk op dat, naast het CREATE OR ALTER PROCEDURE statement, ook het CREATE PROCEDURE statement en het ALTER PROCEDURE statement bestaat. Geen enkele van deze statements kunnen gecombineerd worden met andere statements in een batch.

Stored procedures kunnen alleen aangemaakt worden in de huidige databank.

De stored procedures zijn terug te vinden onder de databanknode, Programmability, Stored Procedures. Rechtsklik op de gewenste stored procedure en kies Modify.

Tip: gebruik het SET NOCOUNT ON statement als eerste statement in de body van de stored procedure. Hierdoor zal SQL Server geen berichten sturen die het aantal rijen tonen dat beïnvloed werd door een Transact-SQL statement of stored procedure. De eliminatie van dit onnodig netwerkverkeer bevordert de performantie.

Maak een stored procedure studentenlijst die de studenten met hun cursussen aflevert.

<pre>create or alter procedure studentenlijst --of: create procedure as set nocount on select s.studnr, s.voornaam, s.familienaam, c.cursusnr, c.cursusnaam from studenten s join studenten_cursussen sc on s.studnr = sc.studnr join cursussen c on sc.cursusnr = c.cursusnr order by s.studnr</pre>
--

Het uitvoeren van een stored procedure gebeurt met het EXEC-statement:

EXEC procedurenaam

Voer de stored procedure studentenlijst uit.
--

exec studentenlijst

Wijzig de stored procedure studentenlijst. Zorg dat <u>alle</u> studenten met hun eventuele cursussen afgeleverd worden.
--

<pre>create or alter procedure studentenlijst --of: alter procedure as set nocount on select s.studnr, s.voornaam, s.familienaam, c.cursusnr, c.cursusnaam from studenten s left join studenten_cursussen sc on s.studnr = sc.studnr left join cursussen c on sc.cursusnr = c.cursusnr order by s.studnr</pre>
exec studentenlijst

Het verwijderen van een stored procedure:

DROP PROCEDURE [IF EXISTS] procedurenaam [, ...]

Verwijder de stored procedure studentenlijst.

drop procedure if exists studentenlijst

System stored procedures zijn SQL Server-ingebouwde stored procedures voor het uitvoeren van administratieve taken. Deze stored procedures zijn gemaakt in de master databank, hebben een prefix sp_ en kunnen vanuit elke databank gebruikt worden. Maak zelf nooit stored procedures met prefix sp_.

3 Data doorgeven aan een stored procedure

Bij het definiëren van een stored procedure kunnen inputparameters opgegeven worden. Elke inputparameter begint met @ en wordt een lokale variabele binnen de procedure. Bij het oproepen van de stored procedure moeten waarden opgegeven worden voor alle parameters, tenzij de parameter een defaultwaarde heeft gekregen.

```
CREATE [OR ALTER] PROCEDURE procedurenaam
@parameter type [=default]
[, ...]
AS
SQL-statement [...]
```

Maak een stored procedure getStudenten met 2 parameters: geboortejaar en geslacht. Deze procedure levert de studenten die geboren zijn na het opgegeven geboortejaar en met het opgegeven geslacht.

```
create or alter procedure getStudenten
@geboortejaar int,
@geslacht char(1)
as
select *
from studenten
where year(geboortedatum) > @geboortejaar and geslacht = @geslacht
exec getStudenten @geboortejaar = 1986, @geslacht = 'V'
exec getStudenten 1986, 'V'
```

Wijzig de stored procedure getStudenten. Deze procedure levert nog steeds de studenten die geboren zijn na het opgegeven geboortejaar en met het opgegeven geslacht. Als er echter géén geboortejaar wordt opgegeven, dan wordt geen rekening gehouden met de geboortedatum en als er geen geslacht wordt opgegeven, dan wordt het geslacht op M gezet.

```
create or alter procedure getStudenten
@geboortejaar int = null,
@geslacht char(1) = 'M'
as
if @geboortejaar is not null
begin
    select *
    from studenten
    where year(geboortedatum) > @geboortejaar and geslacht = @geslacht
end
else
begin
    select *
    from studenten
    where geslacht = @geslacht
end
exec getStudenten
exec getStudenten @geboortejaar = 1986
```

```
exec getStudenten @geslacht='V'  
exec getStudenten @geboortejaar = 1986, @geslacht = 'V'  
exec getStudenten @geboortejaar = 1986, @geslacht = 'M'
```

Maak een stored procedure pasBetaaldAan. Deze procedure zal, voor de opgegeven student, de kolom betaald ophogen met het opgegeven bedrag.

```
create or alter procedure pasBetaaldAan  
@studnr int,  
@bedrag int  
as  
update studenten  
set betaald = betaald + @bedrag  
where studnr = @studnr
```

```
exec pasBetaaldAan @studnr = 1, @bedrag = 50
```

4 Data teruggeven vanuit een stored procedure

Stored procedures kunnen data op 3 manieren teruggeven:

- via een SELECT-statement: zie vorige voorbeelden
- via outputparameters
- via een RETURN-statement.

Bij het definiëren van een stored procedure kunnen outputparameters opgegeven worden met het keyword OUTPUT. Het keyword OUTPUT is ook vereist bij de oproep van de procedure. Binnen de stored procedure gedraagt de outputparameter zich als een lokale variabele. In de oproepende procedure of batch moet een lokale variabele gedeclareerd worden om de waarde op te vangen.

```
CREATE [OR ALTER] PROCEDURE procedurenaam  
@parameter type [=default] [OUTPUT]  
[, ...]  
AS  
SQL-statement [...]
```

Maak een stored procedure getStudentnaam die de voornaam en de familienaam van een student waarvan het studentnummer werd doorgegeven, aflevert onder de vorm van outputparameters.

```
create or alter procedure getStudentnaam  
@studentnr int,  
@voornaam nvarchar(50) OUTPUT,  
@familienaam nvarchar(50) OUTPUT  
as  
set nocount on  
select @voornaam = voornaam, @familienaam = familienaam  
from studenten  
where studnr = @studentnr
```

```
declare @vn nvarchar(50), @fn nvarchar(50)  
exec getStudentnaam 1, @vn OUTPUT, @fn OUTPUT  
select @vn, @fn
```

Het RETURN-statement beëindigt de procedure en levert een geheel getal af aan de oproepende procedure of batch. In de oproepende procedure of batch moet een lokale variabele gedeclareerd worden om de returnwaarde op te vangen. De returnwaarde is vaak een statuscode, maar het kan ook het resultaat zijn van een (gehele) berekening.

Maak een procedure somBetaald. Deze procedure levert het totaal betaalde bedrag van de studenten geboren in een bepaald jaar.

```
create or alter procedure SomBetaald
@jaar int
as
set nocount on
declare @som int
select @som = sum(betaald)
from studenten
where year(geboortedatum) = @jaar
return @som
```

```
declare @res int
exec @res = SomBetaald 1985
select @res
```