

# Stored Procedure & Trigger 2.HF

It & Data, Odense

# Repetition: View

- Et view kan betragtes som en måde at se på data.
- Med et view kan man begrænse antallet af kolonner eller/og antallet af poster, man ser.
- Indholdet af et View dannes først, når forespørgslen laves.
- Virtuel tabel baseret på resultatet af en SQL statement.
- Fordel: Altid adgang til de nyeste data
- Kan ikke ændre i tabeller, kan ikke tage parameter med ligesom i stored procedure
- Eksempel:

```
--Views: Hent alle bøger|
CREATE VIEW vAlleBoeger AS
SELECT b.titel, b.referat, bg.beskrivelse, CONCAT(f.fornavn, ' ', f.etternavn) as forfatter, fl.forlag, b.udgivet
FROM boeger b
INNER JOIN boggenre bg ON b.id_genre = bg.genre_id
INNER JOIN forlag fl ON b.id_forlag = fl.forlags_id
INNER JOIN forfatterbog fg ON b.bog_id = fg.id_bog
INNER JOIN forfatter f ON fg.id_forfatter = f.forfatter_id;
```

# Stored Procedure

- Er ikke andet en gemt kode, som man kan bruge igen og igen.

```
CREATE PROCEDURE spInsertElevData
    @Fornavn NVARCHAR(255),
    @Efternavn NVARCHAR(255),
    @Telefon INT
AS
BEGIN
    INSERT INTO Elev
        (fornavn, efternavn, telefon)
    VALUES
        (@Fornavn, @Efternavn, @Telefon)
END
```

```
-- Verify that the stored procedure does not already exist.
IF OBJECT_ID ( 'usp_GetErrorInfo', 'P' ) IS NOT NULL
    DROP PROCEDURE usp_GetErrorInfo;
GO

-- Create procedure to retrieve error information.
CREATE PROCEDURE usp_GetErrorInfo
AS
SELECT
    ERROR_NUMBER() AS ErrorNumber
    ,ERROR_SEVERITY() AS ErrorSeverity
    ,ERROR_STATE() AS ErrorState
    ,ERROR_PROCEDURE() AS ErrorProcedure
    ,ERROR_LINE() AS ErrorLine
    ,ERROR_MESSAGE() AS ErrorMessage;
GO

BEGIN TRY
    -- Generate divide-by-zero error.
    SELECT 1/0;
END TRY
BEGIN CATCH
    -- Execute error retrieval routine.
    EXECUTE usp_GetErrorInfo;
END CATCH;
```

Læs mere her:

<https://docs.microsoft.com/en-us/sql/t-sql/functions/system-functions-transact-sql?view=sql-server-2017>

# Stored Procedure – mere kompleks

```
-- Stored procedures
--Opret ny låner version 2
CREATE PROCEDURE opretLaanerID
    @Postnr SMALLINT,
    @Bynavn NVARCHAR(100),
    @Fornavn NVARCHAR(125),
    @Efternavn NVARCHAR(50),
    @Gade NVARCHAR(100),
    @Husnr SMALLINT,
    @LaanerID INT OUTPUT
AS
BEGIN
    --Tjek om postnr/by eksistere
    IF (not exists(SELECT * FROM Postby WHERE postnr = @Postnr))
    BEGIN
        INSERT INTO Postby (postnr, bynavn) VALUES (@Postnr, @Bynavn);
        PRINT N'Oprettet nyt postnummer';
    END
    ELSE
        PRINT N'Postnummer er oprettet';

    -- Tjek om en ved samme navn er oprettet
    IF (not exists(SELECT * FROM Laaner WHERE fornavn = @Fornavn and efternavn = @Efternavn))
    BEGIN
        INSERT INTO Laaner (fornavn, efternavn, gade, husnummer, postnr) VALUES (@Fornavn, @Efternavn, @Gade, @Husnr, @Postnr);
        SET @LaanerID = SCOPE_IDENTITY();
        PRINT N'Oprettet ny låner';
    END
    ELSE
    BEGIN
        PRINT N'Låner er oprettet!';
        SELECT @LaanerID = laaner_id FROM Laaner WHERE fornavn = @Fornavn AND efternavn = @Efternavn
    END
    RETURN @@ERROR
END
GO
```

# Stored Procedure

- Hvis kode skal ændres, bruges alter

```
1  USE [myFirstDb]
2  GO
3  /***** Object:  StoredProcedure [dbo].[spInsertElevData]    Script Date: 14-05-2019 08:11:36 *****/
4  SET ANSI_NULLS ON
5  GO
6  SET QUOTED_IDENTIFIER ON
7  GO
8
9  ALTER PROCEDURE [dbo].[spInsertElevData]
10     @Fornavn NVARCHAR(255),
11     @Efternavn NVARCHAR(255),
12     @Telefon INT
13  AS
14  BEGIN
15      INSERT INTO Elev
16      (fornavn, efternavn, telefon)
17      VALUES
18      (@Fornavn, @Efternavn, @Telefon)
19  END
20
21
```

Læs mere på:

<https://www.mssqltips.com/sqlservertutorial/160/sql-server-stored-procedure-tutorial/>

# Trigger

- En trigger er et lille program, der ved en bruger defineret aktivitet udfører en bestemt handling.
- Triggers er meget lig en stored procedure, men bliver kun fyret af ved INSERT, UPDATE eller DELETE.
- Trigger knytter sig til den tabel, der ændres på.
- Man kan f.eks. bruge en triggers, hvis man vil f.eks. sende en mail, hver gang en tabel er blevet opdateret, f.eks. ved en ny ordre.
- Eller hvis man opretter en ny medarbejder, så skal der også oprettes tilsvarende i andre tabeller, som vedr. medarbejderen.

Læs nærmere på:

<https://www.mssqltips.com/sqlservertip/5909/sql-server-trigger-example/>



# Trigger eksempler:

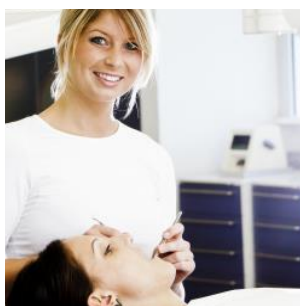
- Simpel trigger, der giver besked ved indsættelse eller opdatering af data:

```
CREATE TRIGGER reminder1
ON Sales.Customer
AFTER INSERT, UPDATE
AS RAISERROR ('Notify Customer Relations', 16, 10);
GO
```

- Trigger, der sender en mail

```
CREATE TRIGGER reminder2
ON Sales.Customer
AFTER INSERT, UPDATE, DELETE
AS
EXEC msdb.dbo.sp_send_dbmail
    @profile_name = 'AdventureWorks2012 Administrator',
    @recipients = 'danw@Adventure-Works.com',
    @body = 'Don''t forget to print a report for the sales force.',
    @subject = 'Reminder';
GO
```

# Slut på Stored Procedure & Triggers



Campus M – Vi uddanner Danmarks dygtigste