-- Tworzenie 3 tabel

-- 2 tabele stacje meteorologiczne i jedna centala

-- stacje meteorologiczne z triggerami wysylajacymi dane do centali

create database stacja\_centrala

go

use stacja\_centrala

go

create table DANE\_GL

(id int primary key identity(1,1),

stacja varchar(20),

data smalldatetime,

temperatura int,

cisnienie int

)

go

CREATE TRIGGER ODPOWIEDZ

ON DANE\_GL

AFTER INSERT

AS

DECLARE @stacja varchar(8),

@data smalldatetime,

@adres varchar(20),

@COS NVARCHAR(100)

SELECT @stacja=stacja,@data=data

FROM INSERTED

SET @COS = N'UPDATE '+rtrim(@stacja)+'.dbo.dane SET stan = ''P'' WHERE stacja=@s and data=@d'

EXEC sp\_executesql @cos, N'@s varchar(8),@d smalldatetime',@s=@stacja, @d=@data

/\* powy¿szy EXEC zamiast poni¿ej

IF @stacja='stacja\_1'

BEGIN

UPDATE STACJA\_1.dbo.DANE

SET stan = 'P'

WHERE stacja=@stacja and

data=@data

END

ELSE

BEGIN

IF @stacja='stacja\_2'

UPDATE STACJA\_2.dbo.DANE

SET stan = 'P'

WHERE stacja=@stacja and

data=@data

END

\*/

GO

-- Tworzenie 1 Stacji: Baza z jedn¹ tabel¹ i triggerem

-- na koñcu wprowadzony 1 wiersz testowy

create database stacja\_1

go

use stacja\_1

go

create table DANE

(id int primary key identity(1,1),

stacja char(20) default('stacja\_1'),

data smalldatetime default (getdate()),

temperatura int,

cisnienie int,

stan char(1) default('F')

)

go

CREATE TRIGGER PO\_WSTAWIENIU\_DANE

ON DANE

AFTER INSERT

AS

DECLARE @stacja varchar(20),

@data smalldatetime,

@temperatura int,

@cisnienie int

SELECT @stacja=stacja,@data=data,@temperatura=temperatura,@cisnienie=cisnienie

FROM INSERTED

INSERT INTO STACJA\_CENTRALA.dbo.DANE\_GL (stacja,data,temperatura,cisnienie)

VALUES (@stacja,@data,@temperatura,@cisnienie)

GO

-- Tworzenie 2 Stacji: Baza z jedna tabela i triggerem

-- na koñcu wprowadzony 1 wiersz testowy

create database stacja\_2

go

use stacja\_2

go

create table DANE

(id int primary key identity(1,1),

stacja char(20) default('stacja\_2'),

data smalldatetime default (getdate()),

temperatura int,

cisnienie int,

stan char(1) default('F')

)

go

CREATE TRIGGER PO\_WSTAWIENIU\_DANE

ON DANE

AFTER INSERT

AS

DECLARE @stacja varchar(20),

@data smalldatetime,

@temperatura int,

@cisnienie int

SELECT @stacja=stacja,@data=data,@temperatura=temperatura,@cisnienie=cisnienie

FROM INSERTED

INSERT INTO STACJA\_CENTRALA.dbo.DANE\_GL (stacja,data,temperatura,cisnienie)

VALUES (@stacja,@data,@temperatura,@cisnienie)

GO

-- sprawdzenie stanu 3 tabel

-- wstawianie przyk³adowych danych do bazy stacja\_1

USE stacja\_1

GO

insert into DANE (temperatura,cisnienie,data) values (10,1000,'2015-11-20 11:00')

insert into DANE (temperatura,cisnienie,data) values (11,1001,'2015-11-20 11:15')

USE stacja\_2

GO

-- wstawianie przyk³adowych danych do bazy stacja\_2

insert into DANE (temperatura,cisnienie,data) values (20,1030,'2015-11-20 11:00')

insert into DANE (temperatura,cisnienie,data) values (21,1035,'2015-11-20 11:15')

GO

SELECT \* FROM stacja\_1.dbo.dane

SELECT \* FROM stacja\_2.dbo.dane

SELECT \* FROM stacja\_centrala.dbo.dane\_gl

/\*

use master

go

drop database stacja\_1

go

drop database stacja\_2

go

drop database stacja\_centrala

\*/

-- usuwanie tylko danych

/\*

use stacja\_1

go

delete from dane

TRUNCATE TABLE [dane];

--DBCC CHECKIDENT ('[dane]', RESEED, 0)

go

use stacja\_2

go

delete from dane

TRUNCATE TABLE [dane];

--DBCC CHECKIDENT ('[dane]', RESEED, 0)

go

use stacja\_centrala

go

delete from dane\_gl

TRUNCATE TABLE [dane\_gl];

--DBCC CHECKIDENT ('[dane\_gl]', RESEED, 0)

go

\*/