

Perspektywy



CREATE VIEW Dane AS

SELECT Nazwisko, Imie FROM Osoby

GO

INSERT INTO Dane VALUES ('Kowalski', 'Wiesław')

INSERT INTO Dane VALUES ('Nowak', 'Jan')

UPDATE Dane SET Imie='Piotr' WHERE Imie= 'Wiesław'

DELETE FROM Dane WHERE Nazwisko= 'Nowak' AND Imie='Jan'

GO


SELECT * FROM Dane

SELECT * FROM Osoby

Widok - perspektywa

Nazwisko	Imie
Kowalski	Jan
...	...
Kowalski	Piotr

IdOsoby	IdDzialu	Nazwisko	Imie	RokUrodz	wzrost
1	1	Kowalski	Jan	1976	1.67
...
15	NULL	Kowalski	Piotr	NULL	NULL



Widok - perspektywa

```
CREATE VIEW Dane_check AS  
SELECT Nazwisko, Imie, Wzrost FROM Osoby  
WHERE Wzrost > 1.70
```

GO

```
INSERT INTO Dane_check VALUES ('Kowalski', 'Karol', 1.8)
```

```
INSERT INTO Dane_check VALUES ('Nowak', 'Jan', 1.6)
```

```
INSERT INTO Dane_check VALUES ('Kowal', 'Piotr', 1.8)
```

```
UPDATE Dane_check SET Wzrost=1.6 WHERE Nazwisko = 'Kowalski' AND Imie=  
'Karol'
```

```
DELETE FROM Dane_check WHERE Nazwisko= 'Nowak' AND Imie='Jan'
```

GO

```
SELECT * FROM
```

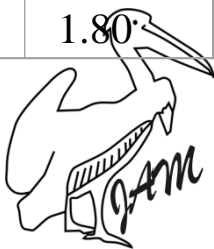
```
Dane_check
```

```
SELECT * FROM Osoby
```

Nazwisk	Imie	Wzrost
Nowak	Karol	1.72
...
Nowak	Edward	1.93
Kowal	Piotr	1.80

IdOsoby	IdDzialu	Nazwisko	Imie	RokUrodz	wzrost
1	1	Kowalski	Jan	1976	1.67
...
15	NULL	Kowalski	Piotr	NULL	NULL
17	NULL	Kowalski	Karol	NULL	1.60
18	NULL	Nowak	Jan	NULL	1.60
19	NULL	Kowal	Piotr	NULL	1.80

**Zezwala na wprowadzenie
danych niezgodnych z
warunkiem
w klauzuli WHERE**



Widok - perspektywa

```
CREATE VIEW Dane_check AS  
SELECT Nazwisko, Imie, Wzrost FROM Osoby  
WHERE Wzrost > 1.70  
WITH CHECK OPTION
```

Nie zezwala na
wprowadzenie
danych niezgodnych z
warunkiem
w klauzuli WHERE

```
GO  
INSERT INTO Dane_check VALUES ('Kowalski', 'Karol', 1.8)  
INSERT INTO Dane_check VALUES ('Nowak', 'Jan', 1.6)  
INSERT INTO Dane_check VALUES ('Kowal', 'Piotr', 1.8)  
UPDATE Dane_check SET Wzrost=1.6 WHERE Nazwisko = 'Kowalski'  
AND Imie= 'Karol'  
DELETE FROM Dane_check WHERE Nazwisko= 'Nowak' AND Imie='Jan'  
GO  
SELECT * FROM Dane_check  
SELECT * FROM Osoby
```

Msg 550, Level 16, State 1, Line 2
The attempted insert or update failed because the target view either
specifies WITH CHECK OPTION or spans a view that specifies WITH CHECK
OPTION and one or more rows resulting from the operation did not qualify
under the CHECK OPTION constraint.
The statement has been terminated.



Widok - perspektywa

```
CREATE VIEW zlaczenie
```

```
AS
```

```
SELECT Nazwa, Nazwisko
```

```
FROM Dzialy JOIN Osoby
```

```
ON Dzialy.IdDzialu= Osoby.IdDzialu
```

```
GO
```

```
INSERT INTO zlaczenie VALUES ('Nowy', 'Nowak')
```

Nie można modyfikować danych przez
perspektywy zawierające wyrażenia lub
odwołujące się do więcej niż jednej tabeli

Msg 4405, Level 16, State 1, Line 1

View or function 'zlaczenie' is not updatable because the
modification affects multiple base tables.



Widok - perspektywa

```
CREATE VIEW aliasy
```

```
(NazwiskoPrac, ImiePrac, RokUrodz)
```

```
AS
```

```
SELECT Nazwisko, Imie, RokUrodz
```

```
FROM Osoby
```

```
GO
```

```
INSERT INTO aliasy(NazwiskoPrac, ImiePrac)
```

```
VALUES ( 'Nowak' , 'Jan' )
```

W definicji perspektywy można określić aliasy kolumn



Widok - perspektywa

```
CREATE VIEW LiczbaWyplat
AS
WITH LiczbaWyplat (Kto, Ile) AS
(
    SELECT IdOsoby, COUNT(*)
    FROM Zarobki GROUP BY IdOsoby)
SELECT Nazwisko, Ile
FROM Osoby JOIN LiczbaWyplat
ON IdOsoby=kto
GO
SELECT * FROM LiczbaWyplat
```

Nazwisko	Ile
KOWALSKI	4
NOWAK	2
KOW	2
JANIK	1
KOWALCZYK	1
NOWICKI	4
ADAMCZYK	1
ZIĘBA	1

W definicji perspektywy można stosować klauzulę WITH.
Jedynym niedopuszczalnym elementem składniowym jest
sortowanie – ORDER BY



Zastosowanie EXISTS do usuwania widoku

IF EXISTS

(SELECT * FROM SYSOBJECTS

WHERE Id = Object_Id(N'Dane_check')

AND OBJECTPROPERTY(Id, N'IsView') = 1)

DROP VIEW Dane_check

GO

IF EXISTS

(SELECT * FROM SYSOBJECTS

WHERE name = 'Dane_check'

AND xtype = 'V')

DROP VIEW Dane_check

GO

IF EXISTS

(SELECT * FROM

INFORMATION_SCHEMA.VIEWS

WHERE TABLE_NAME = 'Dane_check')

DROP VIEW Dane_check

GO




```
SELECT name FROM  
sys.all_objects  
WHERE type= 'V'  
AND schema_id =4  
ORDER BY name
```

service_queues
services
soap_endpoints
spatial_index_tessellations
spatial_indexes
spatial_reference_systems
sql_dependencies
sql_expression_dependencies
sql_logins
sql_modules
stats
stats_columns
symmetric_keys
synonyms
sysaltfiles
syscacheobjects
syscharsets
syscolumns

Perspektywy systemowe właściciela sys

syscomments
sysconfigures
sysconstraints
syscurconfigs
syscursorcolumns
syscursorrefs
syscursors
syscursortables
sysdatabases
sysdepends
sysdevices
sysfilegroups
sysfiles
sysforeignkeys
sysfulltextcatalogs
sysindexes
sysindexkeys
syslanguages
syslockinfo
syslogins
sysmembers
sysmessages



Perspektywy systemowe właściciela sys - przykład

```
SELECT * FROM  
sys.trigger_event_types
```

TYPE	TYPE_NAME	PARENT TYPE NULL
10001	DDL_EVENTS	
10002	DDL_SERVER_LEVEL_EVENTS	10001
10003	DDL_ENDPOINT_EVENTS	10002
10004	DDL_DATABASE_EVENTS	10002
10005	DDL_SERVER_SECURITY_EVENTS	10002
10006	DDL_LOGIN_EVENTS	10005
10007	DDL_GDR_SERVER_EVENTS	10005
10008	DDL_AUTHORIZATION_SERVER_EVENTS	10005
10009	DDL_CREDENTIAL_EVENTS	10005
10010	DDL_SERVICE_MASTER_KEY_EVENTS	10005
10011	DDL_EXTENDED_PROCEDURE_EVENTS	10002
10012	DDL_LINKED_SERVER_EVENTS	10002
10013	DDL_LINKED_SERVER_LOGIN_EVENTS	10012
10014	DDL_MESSAGE_EVENTS	10002
10015	DDL_REMOTE_SERVER_EVENTS	10002
10016	DDL_DATABASE_LEVEL_EVENTS	10001
10017	DDL_TABLE_VIEW_EVENTS	10016
10018	DDL_TABLE_EVENTS	10017
10019	DDL_VIEW_EVENTS	10017
10020	DDL_INDEX_EVENTS	10017



```
SELECT name FROM  
sys.all_objects  
WHERE type= 'V'  
AND schema_id =3  
ORDER BY name
```

Perspektywy systemowe właściciela INFORMATION_SCHEMA

CHECK_CONSTRAINTS
COLUMN_DOMAIN_USAGE
COLUMN_PRIVILEGES
COLUMNS
CONSTRAINT_COLUMN_USAGE
CONSTRAINT_TABLE_USAGE
DOMAIN_CONSTRAINTS
DOMAINS
KEY_COLUMN_USAGE
PARAMETERS
REFERENTIAL_CONSTRAINTS
ROUTINE_COLUMNS
ROUTINES
SCHEMATA
TABLE_CONSTRAINTS
TABLE_PRIVILEGES
TABLES
VIEW_COLUMN_USAGE
VIEW_TABLE_USAGE
VIEWS



Perspektywy INFORMATION_SCHEMA przykłady

```
SELECT * FROM  
INFORMATION_SCHEMA.TABLES
```

TABLE_CATALOG	TABLE_SCHEMA	TABLE_NAME	TABLE_TYPE
test	dbo	BulkTest	BASE TABLE
test	dbo	TXML	BASE TABLE
test	dbo	TAB	BASE TABLE
test	dbo	Pracownicy	BASE TABLE
test	dbo	nnn	BASE TABLE
test	dbo	Dane_check	VIEW
test	dbo	Dane	VIEW
test	dbo	T1	BASE TABLE
test	dbo	sysdiagrams	BASE TABLE
test	dbo	Dzialy	BASE TABLE
test	dbo	Osoby	BASE TABLE
test	dbo	Zarobki	BASE TABLE



Perspektywy INFORMATION_SCHEMA - przykłady

```
SELECT * FROM INFORMATION_SCHEMA.VIEWS
```

TABLE_CATALOG	TABLE_SCHEMA	TABLE_NAME	VIEW_DEFINITION	CHECK_OPTION	IS_UPDATABLE
test	dbo	Dane_check	CREATE VIEW Dane_check AS SELECT Nazwisko, Imie, Wzrost FROM Osoby WHERE Wzrost > 1.70 WITH CHECK OPTION	CASCADE	NO
test	dbo	Dane	CREATE VIEW Dane AS SELECT Nazwisko, Imie FROM Osoby	NONE	NO



Perspektywy INFORMATION_SCHEMA - przykłady

SELECT * FROM

INFORMATION_SCHEMA.TABLE_CONSTRAINTS

CONSTRAINT_CATALOG	CONSTRAINT_SCHEMA	CONSTRAINT_NAME	TABLE_CATALOG	TABLE_SCHEMA	TABLE_NAME	CONSTRAINT_TYPE	...
test	dbo	UK_principal_name	test	dbo	sysdiagrams	UNIQUE	...
test	dbo	PK__Pracowni__7BF2E88057A801BA	test	dbo	Pracownicy	PRIMARY KEY	...
test	dbo	PK_Osoby	test	dbo	Osoby	PRIMARY KEY	...
test	dbo	kl	test	dbo	nnn	PRIMARY KEY	...
test	dbo	fk	test	dbo	nnn	FOREIGN KEY	...
test	dbo	PK__sysdiagr__C2B05B617EF6D905	test	dbo	sysdiagrams	PRIMARY KEY	...



Następnie procedury i funkcje

