-- LABORATORIUM 11.05

-- WYZWALACZE DML

create table T1

(klucz int not null primary key,

dane varchar(10) not null)

create table Tab\_Audit

(klucz\_audit int not null identity(1,1) primary key,

dt datetime2 not null default(sysdatetime()),

login\_name sysname not null default(original\_login()),

klucz int not null,

dane varchar(10) not null)

go

create trigger trg\_T1 on T1

after insert

as

set nocount on

insert into Tab\_Audit(klucz, dane)

select klucz, dane from inserted

insert into T1(klucz, dane) values (10, 'a')

insert into T1(klucz, dane) values (20, 'x')

select \* from Tab\_Audit

drop table Tab\_Audit, T1

--wyzwalacze DDL

create table AuditDDLEvents

(klucz int not null identity(1,1),

posttime datetime2 not null,

eventtype sysname not null,

loginname sysname not null,

schemaname sysname not null,

objectname sysname not null,

targetobjectname sysname null,

eventdata xml not null,

constraint PK\_AuditDDLEvent primary key(klucz))

go

create trigger trg\_audit\_ddl\_events on database for ddl\_database\_level\_events

as

set nocount on

declare @eventdata as xml=eventdata()

insert into AuditDDLEvents(posttime, eventtype, loginname, schemaname, objectname, targetobjectname, eventdata) values(

@eventdata.value('(/EVENT\_INSTANCE/PostTime)[1]','varchar(23)'),

@eventdata.value('(/EVENT\_INSTANCE/EventType)[1]','sysname'),

@eventdata.value('(/EVENT\_INSTANCE/LoginName)[1]','sysname'),

@eventdata.value('(/EVENT\_INSTANCE/SchemaName)[1]','sysname'),

@eventdata.value('(/EVENT\_INSTANCE/ObjectName)[1]','sysname'),

@eventdata.value('(/EVENT\_INSTANCE/TargetObjectName)[1]','sysname'),

@eventdata)

go

create table T1

(col1 int not null primary key)

alter table T1 add col2 int not null

alter table T1 alter column col2 numeric(3)

create nonclustered index idx1 on T1(col2)

select \* from AuditDDLEvents

-- 1. Utworzyć tabelę o nazwie Osoba zawierająca 5 kolumn:

-- identyfikator - liczba sześciocyfrowa, autonumeracja od 100000 co 10

-- imie - tekst 30 znaków

-- nazwisko - tekst 50 znaków

-- wiek - liczba trzycyfrowa

-- data\_dodania - data, domyślnie ma być wstawiana dzisiejsza data

create table osoba(id numeric(6) primary key identity(100000,10),

imie varchar(30),

nazwisko varchar(50),

wiek numeric(3,0),

data\_dodania date default getdate());

-- 2. Do tabeli Osoba dodać 4 wiersze:

-- Jan Kowalski, 35 lat

-- Anna Nowak, 30 lat

-- Ewa Zielińska, 38 lat

-- Adam Woźniak, 32 lata

insert into osoba(imie,nazwisko,wiek)

values

('Jan','Kowalski',35),

('Anna','Nowak',30),

('Ewa','Zielińska',38),

('Adam','Woźniak',31);

select \* from osoba

-- 3. Utworzyć wyzwalacz o nazwie DodanoOsobe na tabeli o nazwie Osoba, który będzie informował użytkownika, że wiersz został dodany. Dodać jeden wiersz do tabeli.

If (OBJECT\_ID('DodanoOsobe') IS NOT NULL)

DROP TRIGGER DodanoOsobe

GO

CREATE TRIGGER DodanoOsobe

ON Osoba

FOR INSERT

AS

PRINT 'Nowa osoba!!!'

insert into osoba(imie,nazwisko,wiek)

values

('Ala','Nowa',28)

-- 4. Usunąć wyzwalacz o nazwie DodanoOsobe.

DROP TRIGGER DodanoOsobe

-- 5. Utworzyć wyzwalacz o nazwie ModyfikujOsobe, który zablokuje możliwość modyfikacji wierszy w tabeli o nazwie Osoba.

-- Wywoływanie instrukcji INSERT na tabeli osoba ma dodatkowo generować wyjątek.

CREATE TRIGGER ModyfikujOsobe on Osoba

for update

as

begin

rollback transaction

raiserror ('Osoby nie można zmodyfikować', 1, 1)

end

update osoba set wiek=20 where imie='Anna'

-- 6. Usunąć wyzwalacz o nazwie ModyfikujOsobe.

DROP TRIGGER ModyfikujOsobe

-- 7. Utworzyć wyzwalacz o nazwie DodanoOsoby na tabeli o nazwie Osoba, który będzie wyświetlał imię i nazwisko nowo dodawanej osoby (wartości).

CREATE TRIGGER DodanoOsobe on Osoba

after insert

as

declare

@imie varchar(30),

@nazwisko varchar(30)

begin

set nocount on

select @imie = imie from osoba

select @nazwisko = nazwisko from osoba

print 'Imie: ' + @imie + ' nazwisko: ' + @nazwisko

end

insert into osoba(imie,nazwisko,wiek) values

('Jan', 'Nowak', 48)

-- 8. Usunąć wyzwalacz o nazwie DodanoOsoby.

DROP TRIGGER DodanoOsobe

-- 9. Utworzyć wyzwalacz o nazwie SprawdzWiek, który zablokuje możliwość wprowadzenia osoby w wieku innymi niż w przedziale 0-120 lat.

CREATE TRIGGER SprawdzWiek on Osoba

after insert

as

declare

@wiek numeric(3)

begin

set nocount on

select @wiek = wiek from osoba

if @wiek < 100 or @wiek > 120

begin

rollback transaction

raiserror('wiek nie znajduje się w przedziale 0-120 lat',16,1)

end

end

-- 10. Dodać nową osobę w wieku 130 lat. Sprawdzić zawartość tabeli.

insert into osoba(imie,nazwisko,wiek)

values

('Marek','Kowalski',130)

select \* from Osoba

-- 11. Usunąć wyzwalacz o nazwie SprawdzWiek.

DROP TRIGGER SprawdzWiek

-- opcjonalne

select definition, type

from sys.sql\_modules as m

join sys.objects as o

on m.object\_id=o.object\_id

-- 12. Zmienić wiek wszystkich osób dodając im 90 lat. Sprawdzić zawartość tabeli.

-- 13. Zmienić wiek o 90 lat osobom, których wiek po zmianie nie przekroczy 120 lat. Sprawdzić zawartość tabeli.

-- 14. Utworzyć wyzwalacz DodajDane, który w momencie dodawania danych będzie wyświetlał zawartość tabeli, zgodną z wiekiem wprowadzanych osób.

-- jeżeli będziemy dodawali osobę w wieku 23 lat, to ma

-- wyświetlić wszystkie osoby w tym wieku, które są w tabeli

-- Przetestować działanie i usunąć trigger.

-- 16. Utworzyć nową tabelę o nazwie grupa, która będzie przechowywała informację o numerach grup i numerach studentów.

-- Po dodaniu nowego wiersza powinna się wyświetlać informacja jaki student (z tabeli osoba) został przypisany do grupy, poprzez wyzwalacz GrupaStudencka.

-- Przetestować działanie i usunąć trigger.

-- Przetestować działanie i usunąć trigger.