# Ćwiczenia

*Procedury składowane*

Poniższe zadania dotyczące procedur składowanych i funkcji w Microsoft SQL Serwerze należy wykonać w kopii bazy danych Northwind.

# Zadanie 1

Utwórz procedurę składowaną wyświetlającą nazwę oraz cenę wszystkich produktów.

CREATE PROC Z1

AS

SELECT ProductName, UnitPrice

FROM Products

GO

EXEC Z1

# Zadanie 2

Utwórz procedurę składowaną wyświetlającą nazwę oraz cenę produktów należących do kategorii, której **nazwa** (nie identyfikator) zostanie podana jako parametr.

create proc z2 (@name varchar(255))

as

select p.ProductName, p.UnitPrice

from Products as p

inner join Categories as c

on p.CategoryID = c.CategoryID

where c.CategoryName = @name;

go

exec z2 N'Produce'

# Zadanie 3

Zmodyfikuj poprzednią procedurę tak, by poprzez parametr wyjściowy (OUTPUT) przekazywała liczbę produktów w wybranej kategorii.

create proc z3

@name varchar(255),

@qty int output

as

select @qty = count(\*)

from Products as p

inner join Categories as c

on p.CategoryID = c.CategoryID

where c.CategoryName = 'Produce';

go

declare @qty int

exec z3 N'Produce', @qty output

print @qty;

# Zadanie 4

Napisz procedurę, która wyświetli nazwę i cenę produktów o maksymalnej cenie w kategorii, której identyfikator podany jest jako parametr procedury.

create proc z4 @id int

as

select top 1 with ties UnitPrice, ProductName

from Products

where CategoryID = @id

order by UnitPrice desc

go

exec z4 1

# Zadanie 5

Utwórz procedurę aktualizującą cenę produktu o podanym numerze ID. Sprawdź w procedurze, czy produkt o takim identyfikatorze istnieje. Jeśli go nie ma, procedura powinna zakończyć działanie bez wykonywania zmian w tabeli.

create proc z5

@id int,

@price money

as

if exists (select 1

from Products

where ProductID = @id)

begin

update Products

set UnitPrice = @price

where ProductID = @id

end;

go

exec z5 1, 10.00;

# Zadanie 6

Utwórz procedurę dopisującą rekord do tabeli Customers. Sprawdź w procedurze, czy klient o podanym identyfikatorze już jest wpisany do tabeli. Jeśli jest, to procedura powinna zakończyć działanie.

create proc z6

@id nchar(5),

@name nvarchar(40)

as

if not exists (select 1

from Customers

where CustomerID = @id)

begin

insert into Customers(CustomerID, CompanyName)

values (@id, @name)

end;

go

exec z6 'TEST', N'Test Sp. z o.o.';

# Zadanie 7

Przed utworzeniem niżej opisanej procedury wykonaj następujące operacje:

Utwórz kopię tabeli Products i nazwij ją Products2:

SELECT \* INTO Products2 FROM Products

Utwórz w tej tabeli klucz główny (wyżej przedstawione zdanie SQL tego nie robi:

ALTER TABLE Products2 ADD PRIMARY KEY (ProductID)

Następnie utwórz procedurę usuwającą produkt o podanym numerze (ProductID) z tabeli Products2.

create proc z7 @id int

as

delete from Products2

where ProductID = @id;

go

exec z7 77;

# Zadanie 8

Utwórz tabelę archiwum (ProductsArchive – będzie to kopia Products2):

SELECT \* INTO ProductsArchive FROM Products2

Usuń rekordy z ProductsArchive:

DELETE FROM ProductsArchive

Usuń automatyczne numerowanie z kolumny ProductID. W tym celu należy usunąć tę kolumnę i dodać ją jeszcze raz – tym razem bez IDENTITY:

ALTER TABLE ProductsArchive DROP Column ProductID -- w celu usunięcia Identity

ALTER TABLE ProductsArchive ADD ProductID INT

Do tabeli ProductsArchive dodaj kolumnę ArchiveID, która będzie pobierała wartości z sekwencji:

CREATE SEQUENCE ProductsArchiveSequence AS INT START WITH 1

ALTER TABLE ProductsArchive ADD ArchiveID INT

DEFAULT NEXT VALUE FOR ProductsArchiveSequence

Napisz procedurę, która usunie dane produktu o podanym jako parametr identyfikatorze (procedura ma usuwać z tabeli Products2) oraz dopisze go do tabeli ProductsArchive.

create proc z8 @id int

as

begin

insert into ProductsArchive (

ProductName, SupplierID, CategoryID, QuantityPerUnit,

UnitPrice, UnitsInStock, UnitsOnOrder, ReorderLevel,

Discontinued, ProductID

)

select ProductName, SupplierID, CategoryID, QuantityPerUnit,

UnitPrice, UnitsInStock, UnitsOnOrder, ReorderLevel,

Discontinued, ProductID

from Products2

where ProductID = @id;

delete from Products2

where ProductID = @id;

end;

go

exec z8 5;

# Zadanie 9

Do tabeli Products2 dodaj kolumnę Login typu NVARCHAR(128) oraz kolumnę Modified typu DATETIME. Napisz procedurę, która zmodyfikuje cenę jednostkową produktu, którego identyfikator jest parametrem procedury. Drugim parametrem ma być nowa wartość ceny. Ponadto procedura powinna wpisać do kolumny Login nazwę loginu osoby, która wprowadziła nową cenę (należy użyć funkcji SUSER\_SNAME()) a do kolumny Modified wpisze datę i godzinę modyfikacji (należy użyć funkcji GETDATE()).

create proc z9

@id int,

@price money

as

if exists (select 1 from Products2 where ProductID = @id)

begin

update Products2

set UnitPrice = @price,

[Login] = suser\_sname(),

Modified = getdate()

where ProductID = @id

end;

go

exec z9 6, 20.00;