

Prawa dostępu do serwera.

Użytkownicy i uprawnienia –DCL.

Archiwizacja baz danych.

Spis treści

Użytkownicy a role	1
Uprawnienia na SQL Server	3
Jak w MS SQL stworzyć użytkownika i nadać prawa do bazy danych	4
Archiwizacja baz danych	6
<i>Różne linki i ciekawostki –niekoniecznie z przedmiotu baz danych 😊</i>	8

Użytkownicy a role¹

Na SQL Server możemy wyróżnić trzy rodzaje ról:

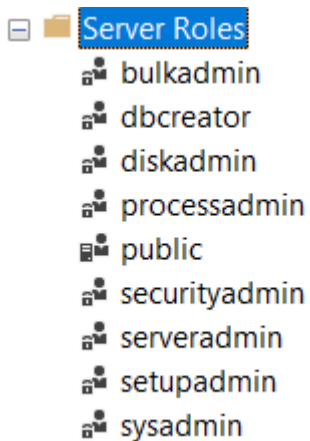
- Serwerowe
- Bazodanowe
- Zdefiniowane przez użytkownika

Role pozwalają grupować użytkowników, którzy mają mieć takie same prawa dostępu. Role serwerowe służą do przyznawania uprawnień operatorom serwera. Role bazodanowe wykorzystuje się do nadania użytkownikom standardowych zasad dostępu do baz danych. Możemy też tworzyć własne role - te role reprezentują najczęściej klasę pracowników w firmie i definiują ich dostęp do aplikacji ulokowanej w konkretnej bazie danych. Takie role znacznie ułatwiają administrację aplikacją biznesową, Np. jeżeli pracownik zmienia stanowisko wystarczy przypisać go do nowej roli i odebrać mu poprzednią, a nie nadawać mu bardzo dużo uprawnień oddzielnie. Dodatkowo jeżeli zmieniają się uprawnienia dla zadanej klasy pracowników wystarczy zmienić uprawnienia dla roli, a nie oddzielnie dla np. tysiąca pracowników.

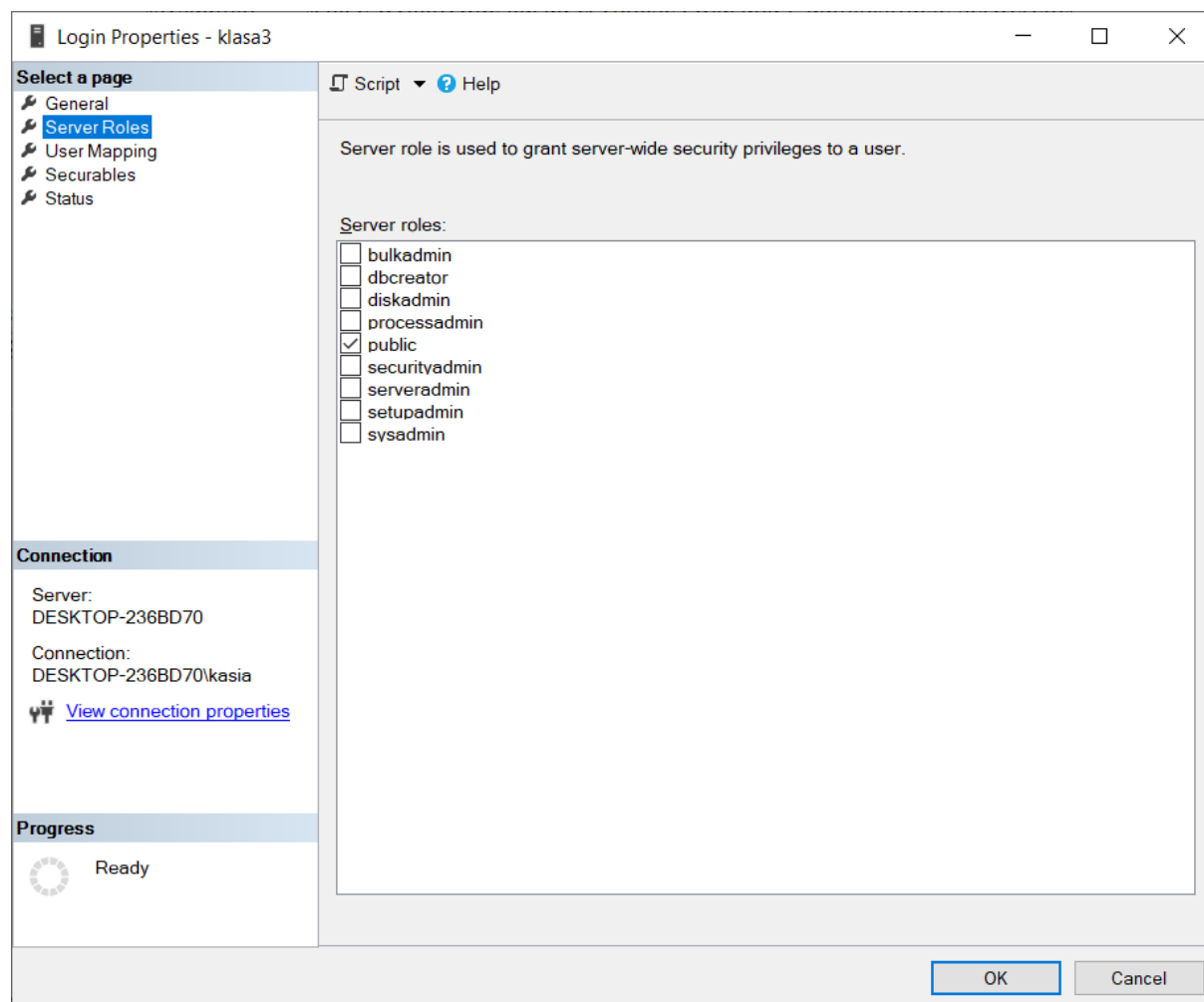
Poniższa tabela zawiera listę wszystkich ról serwerowych

sysadmin	Może wykonywać każdą czynność związaną z administracją serwerem
dbcreator	Może tworzyć i zmieniać ustawienia baz danych
diskadmin	Może zarządzać plikami dyskowymi
processadmin	Może zarządzać procesami SQL Servera
serveradmin	Może konfigurować ustawienia serwera.
setupadmin	Może zainstalować replikację.
securityadmin	Może zarządzać loginami na SQL Serverze.
bulkadmin	Może wykonywać BULK INSERT.

¹ [Microsoft SQL Server \(pjwstk.edu.pl\)](http://Microsoft SQL Server (pjwstk.edu.pl))



Aby przypisać loginowi rolę serwerową, możemy posłużyć się właściwościami loginu (Properties).



Role bazodanowe:

public	Posiada wszystkie publiczne uprawnienia
db_owner	Może przeprowadzić każde działanie na bazie danych
db_accessadmin	Może dodawać, kasować użytkowników baz danych, grupy i role użytkownika
db_ddladmin	Może tworzyć, modyfikować i kasować obiekty baz danych
db_securityadmin	Może zarządzać uprawnieniami systemowymi i obiektowymi
db_backupoperator	Może archiwizować bazę danych
db_datareader	Może czytać z każdej tabeli.
db_datawriter	Może pisać do każdej tabeli.
db_denydatareader	NIE może czytać z żadnej tabeli.

Użytkownikom bazy danych można przypisywać role bazodanowe z poziomu Enterprise Managera - przechodzimy na zakładkę Users wybranej bazy danych i wybieramy Properties danego użytkownika.

Stworzenie własnej roli w bazie na SQL Serverze pozwala na grupowanie użytkowników z identycznymi uprawnieniami. Takie role powinno się tworzyć:

- Kiedy grupa ludzi powinna wykonywać te same czynności na SQL Serverze, a nie istnieje taka grupa w systemie Windows
- Kiedy nie mamy uprawnień do zarządzanie kontami systemu Windows

Grupy zdefiniowane przez użytkownika pozwalają na grupowanie uprawnień i łatwe zarządzanie nimi. Nową rolę możemy utworzyć korzystając z Enterprise Managera - w bazie danych wybieramy zakładkę Roles i wybieramy New.

Uprawnienia na SQL Server

Po zmapowaniu loginów na użytkowników i przypisaniu ich do ról nadszedł czas aby przypisać im uprawnienia bezpośrednio. Uprawnienia przypisuje się tak samo użytkownikom i rolom stworzonym przez użytkownika. Uprawnienia specyfikują z jakich obiektów i co z tymi obiektami mogą robić użytkownicy. Uprawnienia użytkownika w danej bazie danych zależą od uprawnień jakie ma konto tego użytkownika i uprawnienia jakie mają role, do których należy ten użytkownik.

Na SQL Serwerze istnieją dwa rodzaje uprawnień systemowe i obiektowe. Systemowe to np:

- CREATE DATABASE
- CREATE TABLE
- CREATE VIEW
- CREATE PROCEDURE
- CREATE FUNCTION
- CREATE RULE
- CREATE DEFAULT
- BACKUP DATABASE

Uprawnienia obiektowe na tabeli i perspektywie to:

- SELECT
- INSERT
- DELETE
- UPDATE

Uprawnienia obiektowe na kolumnie to:

- SELECT
- UPDATE
- REFERENCES

Uprawnienia obiektowe na procedurze to:

- EXECUTE

Przyznawanie uprawnień

Przyznawanie uprawnień na SQL Serwerze jest trzy stopniowe. Uprawnienia mogą być przyznane (GRANT), zabronione (DENY), albo odebrane (REVOKE). Uprawnienia, które nie zostały nadane, ani zabronione są neutralne - czyli zachowują się jakby były odebrane. Poniższa tabel podsumowuje stany uprawnień.

przyznane - granted	Użytkownik może przeprowadzić zadaną akcję.
zabronione - deny	Użytkownik nie może wykonać zadanej operacji. Ta opcja jest nadrzędna
revoke - odebrane	Użytkownik nie może wykonać zadanej operacji, ale może to być nadpisane poprzez przynależność do roli.

Uprawnienia przyznane (granted) kumulują się więc, czyli użytkownik może wykonać każdą operację jaka została jemu przyznana plus operacja, na które mają uprawnienia role, do których należy. Jak użytkownik ma DENY na jakimś uprawnieniu to to ustawienie nadpisuje wszystkie inne. Użytkownik ma prawo do wykonywania operacji jeśli obydwie poniższe reguły są prawdziwe:

- Uprawnienie zostało nadane bezpośrednio bądź poprzez rolę, do której należy użytkownik.
- Uprawnienie nie zostało zabronione bezpośrednio użytkownikowi, ani żadnej grupie, do której należy.

Jak w MS SQL stworzyć użytkownika i nadać prawa do bazy danych

Do utworzonej bazy Northwind_klasa3... pora dodać użytkownika: klasa3. Aby to uczynić należy wpięrow otworzyć **SQL Server Management Studio** i podłączyć się do wybranej instancji. Następnie trzeba rozwinąć sekcję serwera oraz pozycję **Security**, a następnie kliknąć prawym klawiszem na pozycji **Logins**. W menu podręcznym które zostanie wyświetlone wystarczy wybrać **New Login...** co otworzy okno tworzenia nowego loginu.

The screenshot shows the 'Login - New' dialog box in SQL Server Enterprise Manager. The 'Login name' field contains 'klasa3'. Under 'Authentication', 'SQL Server authentication' is selected. The 'Password' and 'Confirm password' fields are masked with dots. There are checkboxes for 'Specify old password', 'Enforce password policy', 'Enforce password expiration', and 'User must change password at next login'. Below these are options for 'Mapped to certificate', 'Mapped to asymmetric key', and 'Map to Credential'. A 'Mapped Credentials' table is visible with columns 'Credential' and 'Provider'. At the bottom, 'Default database' is set to 'Northwind' and 'Default language' is '<default>'. The 'Connection' section on the left shows 'Server: DESKTOP-236BD70' and 'Connection: DESKTOP-236BD70\klasia'. The 'Progress' section shows a 'Ready' status with a circular progress indicator.

Teraz trzeba podać jego nazwę w polu **Login name** oraz hasło w polach **Password** i **Confirm password**. Można także odznaczyć opcję **Enforce password policy**. Alternatywnie można pozostawić tę opcję zaznaczoną, ale odznaczyć dwie następne: **Enforce password expiration/ wymuszaj wygaśnięcie hasła** oraz **User must change password at the next login**.

Dodatkowo w polu **Default database** należy wybrać bazę danych do której dostęp ma mieć tworzony użytkownik.

Na koniec konieczne jest jeszcze przejście do widoku **User Mapping** gdzie w górnej części okna trzeba zaznaczyć to której bazy użytkownika ma mieć dostęp: **Northwind**. Następnie w dolnej części okna zaznaczyć role **db_datawriter** i **db_datareader**.

The screenshot shows the 'Login - New' dialog box with the 'User Mapping' tab selected. The 'Users mapped to this login:' table is as follows:

Map	Database	User	Default Schema
☐	KSIEGARNIA_ABC		
☐	master		
☐	model		
☐	msdb		
☒	Northwind	klasa3	
☐	ovideo		
☐	tempdb		
☐	TEST1		

Below this table, the 'Guest account enabled for: Northwind' checkbox is checked. The 'Database role membership for: Northwind' list includes the following roles, with 'db_datareader' and 'db_datawriter' selected:

- ☐ db_accessadmin
- ☐ db_backupoperator
- ☒ db_datareader
- ☒ db_datawriter
- ☐ db_ddladmin
- ☐ db_denydatareader
- ☐ db_denydatawriter
- ☐ db_owner
- ☐ db_securityadmin
- ☒ public

The 'Connection' section shows the server 'DESKTOP-236BD70' and connection 'DESKTOP-236BD70\kasia'. The 'Progress' section shows 'Ready'.

W przypadku gdy baza danych będzie używana przez aplikacje używające Entity Framework konieczne jest ustawienie uprawnienia **db_owner** zamiast db_datawriter i db_datareader.

Zadanie1:

Utwórz nowy login o nazwie elektronik z hasłem elektronik. Przypisz bazę np. Northwind do tego loginu i nadaj mu uprawnienie db_owner. Rozłącz się z instancją i połącz się na nowo z na nowym loginie. Co zauważyłeś?

Zadanie 2. Utwórz bazę danych o nazwie klasa3_testowa.

Utwórz użytkownika o nazwie imie haslo imie. Utwórz w niej dowolną tabelę i wstaw do niej 2 dowolne rekordy. Następnie przypisz użytkownikowi imie dostęp do bazy klasa3_testowa (user mapping). Nadaj rolę bazodanową db_datareader , sprawdź na nowym użytkowniku co możesz

wykonać (select, insert update), następnie dołóż rolę sb_datawriter, przetestuj, a na końcu db_ddladmin.

Filmiki do tematu:

- Cz. 1 <https://www.youtube.com/watch?v=mb0PLQZAa2E>
 - Cz. 2 https://www.youtube.com/watch?v=Ei_RmlO6I_4&list=PL07TSwd5SmKZtEUvJ-0k3TNSMh9iO21HC&index=22
-

Archiwizacja baz danych²

Każdy administrator powinien pamiętać o tym jak ważne jest wykonywanie archiwizacji baz danych. Jednymi z powodów są:

- Zabezpieczenie przed przypadkowym wykonaniem instrukcji DELETE bądź UPDATE - np.
- Zabezpieczenie przed wirusami
- Zabezpieczenie przed żywiołami
- Zabezpieczenie przed kradzieżą

Podstawowa zasada dotycząca archiwizacji to : Wykonywać ją regularnie!!

Każda baza danych na SQL Server ma przypisany Recovery Model. W zależności od tego ustawienia mamy różne możliwości dotyczące archiwizacji. Na SQL Server mamy możliwość ustawienia trzech rodzajów Recovery Model:

- Full Recovery
- Bulk Logged
- Simple

Full Recovery tego trybu powinno się używać dla najważniejszych baz w przedsiębiorstwie. W tym trybie SQL Server zapisuje wszystkie zmiany jakie zaszły w bazie danych, łącznie z operacjami typu BULK i tworzeniem indeksów. Dzięki temu SQL Server jest w stanie odtworzyć wszystkie dane do chwili awarii, oprócz transakcji, które akurat trwały podczas awarii. Jedynym minusem tego trybu jest duży rozmiar transactional logu.

Bulk Logged tryb ten podobny jest do Full Recovery, natomiast zużywa on mniej przestrzeni Transactional Log, gdyż nie loguje takich operacji jak tworzenie indeksu, operacje typu BULK, procedure WRITETEXT.

Simple Recovery ten tryb jest dobry dla małych baz danych, w których dane nie zmieniają się za często. Ten typ pozwala odtworzyć tylko dane do ostatniej archiwizacji.

Recovery Model bazy danych można ustawić z poziomu Enterprise Manager - Ustawiamy to we właściwościach bazy danych na zakładce Recovery. Możemy to też ustawić korzystając z instrukcji ALTER DATABASE.

² <https://edu.pjwstk.edu.pl/wyklady/wspbd/scb/adm2/main06.htm>

Przykład:

```
ALTER DATABASE pubs SET RECOVERY simple
```

Archiwizację bazy danych możemy przeprowadzić albo bezpośrednio na dysk albo na "urządzenie". Dla baz produkcyjnych zawsze powinno się przeprowadzać archiwizację na urządzenie. Archiwizację bezpośrednio na dysk powinno się robić w przypadku robienia backupu ad-hoc. Tworząc urządzenie do archiwizacji tworzymy plik, który będzie wykorzystywany do archiwizacji. Urządzenie tworzy się w celu ponownego wykorzystania tego samego pliku i w celu automatyzacji procesu archiwizacji.

Do stworzenia takiego urządzenia możemy posłużyć się procedurą `sp_addumpdevice`.

Przykład:

```
sp_addumpdevice 'disk', 'mojeUrządzenie', 'd:\MSSQLBACKUP\pubsBackup.bkf'
```

Do przeprowadzenia pełnej archiwizacji bazy danych możemy skorzystać z EM - zaznaczamy bazę danych i wybieramy BACKUP DATABASE, albo posłużyć się instrukcją BACKUP DATABASE.

Przykład - archiwizacja bazy pubs na urządzenie:

```
BACKUP DATABASE pubs TO mojeUrządzenia
```

Przykład - Archiwizacja bazy pubs bezpośrednio na dysk:

```
BACKUP DATABASE pubs TO DISK 'd:\backup\pubsBackup.bak'
```

Do odtworzenia bazy danych możemy posłużyć się albo EM - Po zaznaczeniu bazy danych wybieramy RECOVER DATABASE. Albo możemy posłużyć się instrukcją RESTORE DATABASE.

Przykład:

```
RESTORE DATABASE pubs FROM mojeUrządzenie
```

Przykład:

```
RESTORE DATABASE pubs FROM DISK 'd:\backup\pubsBackup.bkf'
```

Różne linki i ciekawostki –niekoniecznie z przedmiotu baz danych ☺

- Wersje porównanie serwerów bazodanowych: <https://www.youtube.com/watch?v=SperI96edtA>
- Baza adventureworks: <https://docs.microsoft.com/en-us/sql/samples/adventureworks-install-configure?view=sql-server-ver15&tabs=ssms>
- Filmiki do bazy adventure works: <https://www.youtube.com/channel/UCDcDG1sE1E8IODEscfRGajA/videos>
- Sql: <http://anonco.pl/sql/#sqlkurs>
- Temat: Backup - bazy MSSQL, system kopii zapasowej: <https://www.youtube.com/watch?v=Ta-PR-tCWa8>
- Różne ms sql: https://www.youtube.com/playlist?list=PLLA-qWsEu6aG6lHk59Fge_GvXtinyuedu
- [beitadminpl - YouTube](#)
- [WebCrew Tomasz Jaśniewski - YouTube](#)
- Access: https://www.youtube.com/channel/UCvIEAU_FPIhFfChDfNIUOkQ/videos