



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Projekt „NEW-TECH Program rozwoju praktycznych kompetencji nauczycieli zawodów branż nowych technologii” jest współfinansowany przez Unię Europejską w ramach środków Europejskiego Funduszu Społecznego

„ Microsoft SQL Server 2012 Express Edition”

MATERIAŁY SZKOLENIOWE

Człowiek – najlepsza inwestycja

www.kapitalludzki.gov.pl

Podręcznik jest współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego.

Podręcznik jest dystrybuowany bezpłatnie.

www.newtech.eduportal.pl

Gdynia 2013



Projekt realizowany przez COMBIDATA Poland sp. z o.o. w ramach umowy o dofinansowanie projektu w ramach Programu Operacyjnego Kapitał Ludzki 2007-2013, Priorytetu III „Wysoka jakość systemu oświaty”, Działanie 3.4 „Otwartość systemu edukacji w kontekście uczenia się przez całe życie”, Poddziałanie: 3.4.3 „Upowszechnienie uczenia się przez całe życie - projekty konkursowe”.

Spis treści

1.	Zarządzanie serwerem Microsoft SQL Server 2012 Express Edition	5
1.1.	Ustawienie dostępu za pomocą protokołu TCPIP	5
1.2.	Nawiązywanie połączenia do serwera za pomocą programu SSMS.....	5
1.3.	Lista baz na serwerze	6
1.4.	Loginy Windows i SQL	6
1.5.	Uprawnienia nadawane za pomocą ról serwera	6
1.6.	Tworzenie użytkowników oraz role bazy danych, uprawnienia w bazie.....	7
1.7.	Tworzenie bazy danych za pomocą kreatora.....	8
1.8.	Wykonywanie backupu pełnego bazy i odzyskiwanie bazy z backupu.....	8
1.9.	Montowanie bazy danych.....	9
1.10.	Tworzenie bazy za pomocą skryptu	9
1.11.	Typy danych	9
1.12.	Tworzenie tabel	11
1.13.	Wstawianie danych do tabeli, ich modyfikacja i usuwanie	11
1.14.	Importowanie danych z tabeli Microsoft Excel	12
1.15.	Pobieranie danych	12
1.16.	Pobieranie danych do Microsoft Excel	13
1.17.	Zapytania do danych geograficznych (OPCJONALNE).....	13
1.18.	Wykonywanie raportów (OPCJONALNE)	14
2.	Ćwiczenia praktyczne.	15
2.1.	Ustawienie dostępu za pomocą protokołu TCPIP.....	15
2.2.	Nawiązywanie połączenia do serwera za pomocą programu SSMS.....	15
2.3.	Lista baz na serwerze	15
2.4.	Loginy Windows i SQL	16
2.5.	Uprawnienia nadawane za pomocą ról serwera	20
2.6.	Tworzenie użytkowników oraz role bazy danych i uprawnienia w bazie	22
2.7.	Tworzenie bazy danych za pomocą kreatora.....	22

2.8.	Wykonywanie backupu pełnego bazy i odtwarzanie bazy z backupu.....	23
2.9.	Montowanie bazy danych	23
2.10.	Tworzenie bazy za pomocą skryptu.....	25
2.11.	Typy danych	30
2.12.	Tworzenie tabel	30
2.13.	Wstawianie danych do tabeli, ich modyfikacja i usuwanie	30
2.14.	Importowanie danych z tabeli Microsoft Excel	30
2.15.	Pobieranie danych	32
2.16.	Pobieranie danych do Microsoft Excel	35
2.17.	Zapytania do danych geograficznych (OPCJONALNE)	41
2.18.	Wykonywanie raportów (OPCJONALNE)	42
3.	Praktyczny przykład zastosowania	43
2.1.	Case Study Projekt bazy danych dla dziennika elektronicznego	43
4.	Załączniki	43
5.	Literatura i wykaz linków.....	43

1. Zarządzanie serwerem Microsoft SQL Server 2012 Express Edition

1.1. Ustawienie dostępu za pomocą protokołu TCPIP

Oprócz Microsoft SQL Server jest dostępnych wiele innych relacyjnych silników bazodanowych. Do najpopularniejszych należy Oracle (jest bezpłatna wersja Express ograniczona do jednej bazy o rozmiarze max 4GB) oraz silniki na bardziej liberalnej licencji (zwykle bezpłatnej): MySQL bardzo popularny w tworzeniu stron internetowych, kompatybilny z nim MariaDB (utworzony przez społeczność tworzącą MySQL po kupieniu go przez Oracle), PostgreSQL- najbardziej zaawansowany RDB na wolnej licencji.

Popularne wersje to Microsoft SQL Server 2000, Microsoft SQL Server 2005, Microsoft SQL Server 2008, Microsoft SQL Server 2008R2, Microsoft SQL Server 2012. W kolejnych wersjach wprowadzano nowe funkcjonalności. Największa zamiana nastąpiła przy przejściu z wersji 2000 na 2005. W ramach każdej z wersji jest dostępnych kilka edycji SQL różniących się funkcjonalnością i ceną. Najbardziej rozbudowana wersja (bardzo droga nawet w przypadku licencji edukacyjnej) to Enterprise dedykowana do dużych rozwiązań obsługujących bazy o rozmiarze kilkuset GB, a nawet wielu TB. Wersja Standard dedykowana do średnich baz ma zablokowane niektóre funkcjonalności niezbędne przy bardzo dużych bazach, ale rzadko używane gdy baza jest mniejsza, nawet jeśli jest umieszczona na wersji Enterprise. Jest również dostępna bezpłatna Express Edition. Jest ona ograniczona funkcjonalnie – maksymalny rozmiar plików bazy to 10 GB (wersja 2008R2 i 2012, wersje 2005 i 2008 to 4GB, a MSDE 2000 to 2GB), ale na jednej instancji można utworzyć wiele baz, każda o rozmiarze 10GB. Edycja Express jest ograniczona wydajnościowo: maksymalna ilość RAM to 1GB i możliwe jest użycie tylko jednego procesora.

Na jednym serwerze można zainstalować wiele instancji Microsoft SQL Server. Instancje to oddzielne serwery z bazami danych, każda z instancji jest całkowicie niezależna od pozostałych. Dlatego każda z nich może mieć inną wersję i edycję Microsoft SQL Server. Jedną z instancji to instancja domyślna, jej nazwa jest taka sama jak nazwa komputera Windows, a usługa nazywa się MSSQLSERVER. Instancja domyślna ma przypisany numer portu 1433. Dodatkowe instancje to instancje nazwane, ich nazwa ma postać NazwaKomputera\NazwaInstancji, a usługa ma nazwę MSSQL\$NazwaInstancji i ma losowo przypisany numer portu (zwykle powyżej 40000).

Po instalacji serwer bazy danych dostępny jest przez pamięć współdzieloną (ang. *Shared Memory*), ale klient musi być na tym samym komputerze, co serwer. Dostęp do instancji serwera SQL można uzyskać z klientów na innych komputerach po włączeniu obsługi TCP/IP i zdefiniowaniu wyjątku na FireWall.

1.2. Nawiązywanie połączenia do serwera za pomocą programu SSMS

Podstawowym klientem Microsoft SQL Server jest SQL Server Management Studio. Ma on szereg funkcjonalności ułatwiających korzystanie z baz danych i zarządzanie serwerem SQL. Po jego uruchomieniu nawiązywane jest połączenie z serwerem. Przy nawiązywaniu należy wpisać nazwę instancji serwera SQL np. Komputer\SQL2012Express. Dla lokalnego komputera zamiast

nazwy można wpisać local lub . (kropka), nazwa instancji może być np. .\SQL2012Express.

Po wpisaniu należy wybrać sposób logowania: SQL lub Windows. Logowanie Windows wykorzystuje wcześniejsze zalogowanie się użytkownika do domeny. Logowanie SQL wymaga podania nazwy loginu (nazwy użytkownika) i hasła.

1.3. Lista baz na serwerze

Po zainstalowaniu Microsoft SQL Server są utworzone bazy systemowe:

master – najważniejsza baza, w której są przechowywane informacje o loginach, zewnętrznych źródłach danych i wszystkich pozostałych bazach danych na serwerze (w bazie master jest informacja, gdzie znajduje się pierwszy plik każdej z baz danych (z rozszerzeniem .mdf), który zawiera wszystkie informacje o bazie danych)

model – jest to szablon bazy danych, gdy stworzymy nową bazę jest ona kopią bazy model, oczywiście część ustawień może zostać zmieniona w czasie tworzenia bazy lub później.

msdb – baza usługi SQL Server Agent, która pozwala na automatyzację działań administracyjnych, w wersji EXPRESS usługa jest zablokowana.

tempdb – baza, w której tworzone są przez użytkownika lub system obiekty tymczasowe, np. w celu szybszego wykonania zapytania, albo obiekty pomocnicze.

bazy użytkownika – bazy utworzone przez użytkownika np. Northwind, AdventureWorks, Spatial, Dziennik. W tych bazach umieszczane są tablice z danymi i inne obiekty, z których korzystają różne aplikacje (np. strony internetowe).

1.4. Loginy Windows i SQL

Użytkownik (lub aplikacja) łącząc się z serwerem bazodanowym musi się zalogować.

W Microsoft SQL Server można tworzyć loginy SQL, gdzie podawany jest login i hasło. Logowanie się na login SQL wymaga ustawienia instancji w tryb MIX. Można tworzyć loginy będące mapowaniem domenowego (lub lokalnego) konta użytkownika Windows albo domenowego (lub lokalnego) konta grupy Windows.

W przypadku logowania Windows do SQL Server, wykorzystywany jest żeton dostępu tworzony przy logowaniu do domeny lub lokalnego systemu.

1.5. Uprawnienia nadawane za pomocą ról serwera

Przypisanie użytkownikowi uprawnień w instancji serwera SQL najlepiej wykonać dodając go do jednej z predefiniowanych ról serwera. Rola **public** jest rolą o minimalnym zestawie uprawnień, każdy tworzony login jest domyślnie dodawany do tej roli. Ma ona uprawnienie połączenia do instancji SQL Serwera. Rolą o najwyższych uprawnieniach jest **sysadmin**. Dodanie loginu do tej roli nadaje wszystkie uprawnienia administracyjne do instancji SQL serwera, we wszystkich bazach systemowych i bazach użytkownika. Rola **dbcreator** ma uprawnienie tworzenia nowych baz

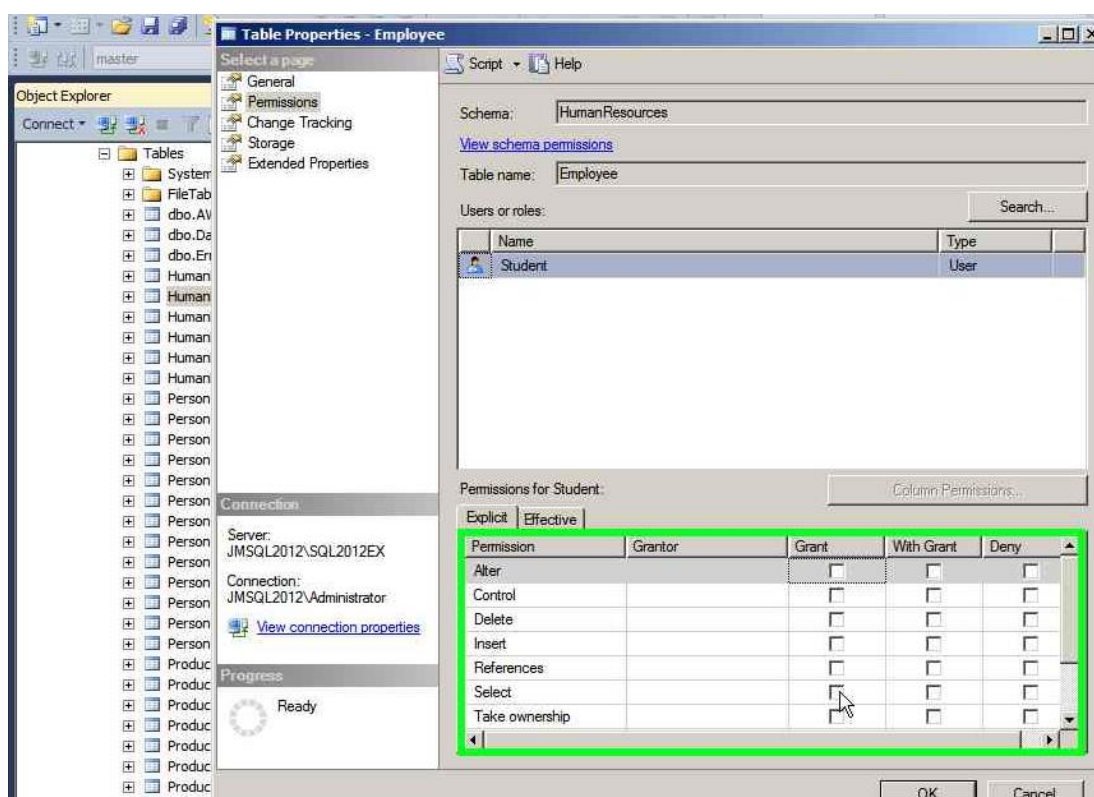
danych. Użytkownik, który utworzył bazę staje się jej właścicielem i ma do niej pełne uprawnienia. Pozostałe role mają przypisane specyficzne uprawnienia administracyjne.

1.6. Tworzenie użytkowników oraz role bazy danych, uprawnienia w bazie

Do loginu utworzonego w instancji serwera można przypisać użytkownika w bazach danych. Użytkownik w każdej z baz danych jest więc mapowany na odpowiadający mu login. Uprawnienia w bazie można przypisywać użytkownikom bezpośrednio do obiektów np. tablic lub poprzez dodanie użytkownika do odpowiedniej roli bazy danych. Użytkownik utworzony w bazie danych jest domyślnie dodawany do roli **public**. Rola **db_datareader** ma uprawnienie czytania z wszystkich tablic w bazie danych, rola **db_datawriter** ma uprawnienie zapisu (wstawiania nowych wierszy, modyfikowania oraz usuwania istniejących wierszy) do wszystkich tablic, rola **db_owner** ma pełne uprawnienia w bazie danych. Dodanie użytkownika do roli **db_denydatareader** uniemożliwia mu odczyt z wszystkich tablic, dodanie użytkownika do roli **db_denydatawriter** zabrania mu zapisu do wszystkich tablic w bazie.

Użytkownikowi można również nadać uprawnienia bazpośrednio.

- Kliknij prawym przyciskiem myszy na obiekcie, do którego chcesz nadać uprawnienia.
- Wybierz zakładkę **Permissions** w oknie **RodzajObiektu Properties - NazwaProjektu**



- Przycisk **Search** pozwala wybrać użytkownika (lub rolę) w sekcji **Permission for ...** (na zrzucie powyżej **Permissions for Student**)

Można zaznaczyć uprawnienia (analogicznie jak loginowi na poziomie serwera). Zestaw uprawnień

zależy od rodzaju obiektu, np. procedury mają uprawnienie EXECUTE (wykonaj), a tablice i widoki uprawnienia SELECT (pobierz, wybierz), INSERT (wstaw), UPDATE (zmień, uaktualnij), DELETE (usuń).

Tworząc użytkownika w bazie można przypisać mu domyślny schemat (ang. *schema*). W małych bazach wszystkie obiekty umieszcza się zazwyczaj w domyślnym schemacie dbo. Jeśli baza zawiera wiele obiektów tworzy się dodatkowe schematy (tak jak dodatkowe podfoldery w Moje Dokumenty).

Każdy obiekt ma czteroczęściową nazwę, gdzie każda część jest oddzielona kropkami:

serwer.baza.schemat.obiekt

Jeśli obiekt (np. tabela) znajduje się na serwerze, do którego jesteśmy podłączeni, w bieżącej bazie i w domyślnym schemacie dbo (ewentualnie w schemacie domyślnym dla usera w danej bazie) można użyć jednoczęściowej nazwy (obiekt). Zalecane jest używanie dwuczęściowych nazw (schema.obiekt), jest to konieczne w przypadku funkcji użytkownika i gdy obiekt jest w innej schemacie niż domyślny. Jeśli obiekt jest w innej bazie należy użyć trzyczęściowej nazwy (baza.schemat.obiekt). Gdy do obiektu łączymy się za pomocą linked server wskazującym zdalne źródło, należy użyć nazwy czteroczęściowej (niektóre bazy nie obsługują wszystkich części nazwy, np. bazy JETa, Access mają postać serwer...tabela, a Excel postać serwer...\$Arkusz).

1.7. Tworzenie bazy danych za pomocą kreatora

W Microsoft SQL serwer można utworzyć bazę za pomocą kreatora uruchamianego z SSMS.

1.8. Wykonywanie backupu pełnego bazy i odzyskiwanie bazy z backupu

Po utworzeniu bazy danych należy wykonywać jej backup. Najczęściej wykonywany jest backup do pliku na dysku lokalnym. Podstawowym i zawsze pierwszym wykonywanym backupem bazy jest backup pełny (Full). W razie awarii za jego pomocą można odtworzyć bazę do stanu w momencie jego wykonania. Backup może być odtworzony na tej samej lub innej instancji serwera, ale często backupu z wyższej edycji nie można odtworzyć na niższej, np. backupu bazy o rozmiarze 11GB z serwera Standard nie można odtworzyć na instancji Express Edition. Bazy ustawionej na poziom SQL 2012 nie można odtworzyć na instancji SQL 2008.

Przy wykonywaniu backupu pełnego kopiowana jest cała baza danych, dlatego przy większych bazach danych uzupełnia się go backupem przyrostowym (Differential). Backup przyrostowy kopiuje tylko te dane, które zmieniły się od ostatniego backupu pełnego. Przy odzyskiwaniu bazy po awarii należy odtworzyć ostatni backup pełny i ostatni wykonany po nim backup przyrostowy. Jeśli chcemy mieć możliwość odzyskania bazy do dowolnego punktu czasowego należy ustawić model odzyskiwania na full i po backupie pełnym (lub pełnym i przyrostowym) wykonywać backup logu (log). Odzyskiwanie po awarii polega na próbie wykonania backupu logu (tzw. końcówka logu czyli tail log), co czasami pomimo uszkodzenia bazy jest możliwe. Następnie odtwarzamy ostatni backup pełny (lub ostatni pełny i ostatni przyrostowy), następnie odzyskujemy wszystkie backupy

logów wykonane od ostatniego backupu pełnego (lub ostatniego przyrostowego) wraz z backupem końcówki logu, jeśli udało się ją wykonać po awarii. Istnieje możliwość zatrzymania odtwarzania logu transakcyjnego w punkcie czasowym (data, godzina, minuta, sekunda ...). Włączenie modelu odzyskiwania pełnego ma jednak pewne konsekwencje. Log transakcyjny ma coraz większy rozmiar i jest on czyszczony po wykonaniu backupu, jeśli w tym modelu nie robimy często backupu, to rozmiar logu może wielokrotnie przekroczyć rozmiar bazy. W modelu odzyskiwania Simple, log jest automatycznie czyszczony, więc jeśli możemy zaakceptować utratę danych od ostatniego backupu pełnego lub przyrostowego należy wybrać model odzyskiwania Simple, ponieważ utrzymanie bazy wymaga mniejszego nakładu pracy administracyjnej.

1.9. Montowanie bazy danych

Utworzona baza danych użytkownika może zostać odmontowana w instancji serwera, następnie po skopiowaniu może zostać zamontowana z innej lokalizacji w tej samej lub innej instancji. Jeśli możemy zaakceptować niedostępność bazy przez czas jej odmontowania i kopiowania działanie to jest szybsze niż wykonanie backupu i jego odzyskanie, które jednak powodują brak dostępności bazy, choć spowalniają ten dostęp poprzez znaczne obciążenie serwera. Każda baza danych składa się przynajmniej z dwóch plików: pierwszego pliku bazy z rozszerzeniem **mdf** oraz logu transakcyjnego z rozszerzeniem **ldf**. Jednak baza danych może mieć dodane dodatkowe pliki bazy z rozszerzeniem **ndf** oraz dodatkowe pliki logów z rozszerzeniem **ldf**. W pierwszym pliku bazy danych (z rozszerzeniem **ndf**) podane są lokalizacje wszystkich pozostałych plików bazy (dodatkowych plików z rozszerzeniem **ndf** i logów z rozszerzeniem **ldf**). Lokalizacja pierwszego pliku bazy jest rejestrowana w bazie **master**. Jeśli położenie względne dodatkowych plików bazy i plików logów nie zmieniło się (np. są w jednym folderze, tylko skopiowanym w inne miejsce na dysku lub na inny dysk) przy montowaniu bazy wystarczy wskazać plik **mdf**, jeśli jednak względne położenie plików się zmieniło (np. każdy skopiowano na inny dysk) może być potrzebna wskazania lokalizacji tych plików.

1.10. Tworzenie bazy za pomocą skryptu

Bazę danych i wszystkie obiekty w niej oraz dane można utworzyć za pomocą skryptu SQL, Dla utworzonego obiektu (bazy, widoku, procedury, tabeli itd.) można wygenerować skrypt.

1.11. Typy danych

W Microsoft SQL Serwer obsługiwanych jest wiele typów danych, jeśli jest taka potrzeba można oprogramować (np. w C#) własne typy danych i po utworzeniu odpowiedniej biblioteki i zamontowaniu jej w bazie jako assembly, utworzyć własny typ danych. Typy danych można podzielić na: liczbowe, finansowe, daty i czasu, tekstowe, binarne, itd.

Do typów liczbowych zaliczamy:

bit – wartości 0 i 1;

tinyint – liczby całkowite od 0 do 255;

smallint – liczby całkowite od -32 768 do 32 767;

int – liczby całkowite od -2 147 483 648 do 2 147 483 648;

bigint – liczby całkowite od -2^{63} do $2^{63}-1$;

float i **real** – duże liczby (do 40 cyfr);

numeric i **decimal** – liczby z ułamkiem dziesiętnym;

money – duże kwoty pieniężne (od 2^{63} do $2^{63}-1$);

smallmoney – pieniądze normalne kwoty (od - 214 768.3648 do 214 768.3647).

Typy daty i czasu to:

smalldatetime – daty od 1 stycznia 1900 roku do 2079 roku z dokładnością do 1 minuty;

datetime – daty od 1753 do 9999 roku z dokładnością do 3.33 milisekundy;

time – czas godzina:minuta:sekunda: część sekundy (ustawiana do 0,000001 sekundy);

date – data od 1 roku n.e do 9999 roku, rok-miesiąc-dzień (bez godzin, minut...);

datetime2 – jak datetime tylko od 1 roku n.e.;

datetimeoffset – oprócz daty (jak w datetime) podane przesunięcie strefy czasowej.

Typy znakowe:

char – stała szerokość, jeśli tekst krótszy, jest uzupełniany znakami białymi;

varchar – zmienna długość (max 8000);

text – duże dokumenty tekstowe (do 2GB), w tabeli 16 bajtowy adres położenia dokumentu na stosie;

varchar(max) – obsługuje duże dokumenty, jeśli rozmiar mniejszy od 8000 znaków przechowywane w tabeli, jeśli ponad 8000 znaków, ale mniej niż 2GB przechowywane jak text, dokumenty większe niż 2GB mogą być przechowywane w FileStream, czyli specjalnym folderze na dysku. Rozmiar dokumentów nie jest wliczany do rozmiaru bazy, co ma szczególne znaczenie przy edycji Express, gdzie rozmiar bazy jest ograniczony do 10GB.

Typy znakowe mają dodatkowo określone **collate**, czyli stronę kodową (np. klawiatura polska maszynisty) oraz dodatkowe ustawienia np. istotna wielkość liter. **Collate** może być dziedziczony z poziomu serwera, bazy danych lub przypisany do kolumny.

Jeśli w jednej kolumnie zachodzi konieczność przechowywania znaków z różnych języków można użyć UNICODE. Wtedy każdy znak jest definiowany dwubajtowym kodem (zamiast jednobajtowego), co daje ponad 64 tys. znaków (zamiast 255 w klasycznym, jednobajtowym kodowaniu). Typy danych UNICODE mają na początku nazwy np. nchar, nvarchar, ntext. Dla typów nchar i nvarchar można przypisać do 4000 znaków. W przypadku typu UNICODE collate określa sposób sortowania np. w collate Latin_ litera ą będzie po z, a w collate Polish litera ą będzie po a, ale przed b. Ci w nazwie collate, określa, że są rozróżniane małe i wielkie litery.

Dane binarne pozwalają na przechowywanie kodu programu, zawartości pliku np. jpg, mp3:

binary – dane binarne o stałym rozmiarze;

varbinary – dane binarne o rozmiarze zmiennym do 8000 znaków;

image – duże dane binarne do 2GB, w tabeli jest 16 bajtowy adres położenia obiektu na sterpie;

varbinary(max) – analogicznie jak varchar(max) i nvarchar(max).

Typy spatials:

geometry – pozwala na tworzenie rysunków na płaszczyźnie złożonych z punktów, łamanych i wieloboków;

geography – pozwala na tworzenie rysunków (map i planów) odnośnie do współrzędnych geograficznych (długość i szerokość) z uwzględnieniem krzywizny Ziemi.

XML – typ ten pozwala na przechowywanie sformalizowanych dokumentów XML. Mają one określoną strukturę i są nazywane samo opisującymi. Są obecnie jednym z najważniejszych standardów wymiany dokumentów.

1.12. Tworzenie tabel

Tabelę w bazie można utworzyć za pomocą graficznego kreatora lub za pomocą skryptu (CREATE TABLE ...). Każda tabela ma unikalną nazwę w bazie danych (może być również dwu częściowa, pełna nazwa jest czteroczęściowa: serwer.baza.schemat.tabela)

W tabeli są kolumny. Każda kolumna ma unikalną w bazie nazwę i określony typ danych. Dodatkowo kolumna może być zdefiniowana jako kolumna umożliwiająca wstawianie wartości pustych (domyślnie NULL lub NOT NULL). Jedna z kolumn może tworzyć klucz podstawowy (w wyjątkowych przypadkach stosuje się klucze złożone z więcej niż jednej kolumny). Zwyczajowo jest przyjęte, że klucz podstawowy ma nazwę składającą się z nazwy tabeli i sufiksu ID. Kolumna podstawowego klucza sztucznego może mieć włączone auto numerowanie, czyli IDENTITY. Kolumna z sufiksem ID o nazwie innej niż tabela, w której się znajduje, oznacza zazwyczaj, że jest to klucz obcy wskazujący na klucz podstawowy w innej tabeli.

1.13. Wstawianie danych do tabeli, ich modyfikacja i usuwanie

Do tabeli można wstawiać dane, modyfikować je i usuwać za pomocą formularzy aplikacji webowej lub okienkowej powiązanej z bazą danych, ale można również bezpośrednio za pomocą klienta np. SSMS (SQL Server Management Studio). SSMS posiada graficzny interfejs (Microsoft Query podobny do znanego wielu osobom z programu Access i Excel). Podstawową metodą wstawiania wierszy do bazy jest napisanie polecenia w oknie query z wykorzystaniem polecenia SQL INSERT. Dane modyfikowane są poleceniem UPDATE, usuwane poleceniem DELETE (pisania zapytań TSQL można nauczyć się na oddzielnym szkoleniu). Zestawy poleceń TSQL można zapisać jako skrypt – plik z rozszerzeniem .sql, który można później otworzyć i wykonać w oknie zapytania SSMS. Do wykonywania takich (i innych) działań administrator baz danych (dba) lub programista może przygotować obiekt zwany procedurą składowaną (stored procedure, sp). Aby użyć procedury składowanej użytkownik podaje nazwę procedury, wymagane lub opcjonalne parametry, nie musi znać całej, często skomplikowanej składni zapytania ukrytego w procedurze składowanej.

1.14. Importowanie danych z tabeli Microsoft Excel

Innym sposobem wstawienia danych do tabeli sql jest wykorzystanie istniejącej tabeli np. utworzonej w programie Microsoft Excel. W SSMS istnieje narzędzie import-export, które pozwala wskazać źródło (np. tabela w arkuszu programu Excel), następnie cel (np. tabela w bazie Microsoft SQL Serwer). Jeśli kolumny źródła i celu mają zgodne typy danych to są one automatycznie mapowane, jeśli nie to w prostym formularzu należy wskazać, z której kolumny tabeli źródłowej do której kolumny tabeli docelowej dane mają zostać przekopiowane. Należy pamiętać, aby w przypadku 64-bit SQL wybierać narzędzie 32 bit import-export, ponieważ wersja 64 bit nie obsługuje baz Jet, do których zaliczane są pliki programu Excel (również Access i pliki tekstowe csv). Oczywiście analogicznie można wyeksportować dane do pliku programu Excel.

1.15. Pobieranie danych

Aby pobrać dane z tabeli należy wykonać zapytanie TSQL z wykorzystaniem polecenia SELECT (pisanie zapytań TSQL można nauczyć się na innych szkoleniach). W programie SSMS klikając prawym przyciskiem myszy na tabelę można wybrać polecenie tworzenia zapytania SELECT, które otwarte i uruchomione w oknie query pobierze wszystkie dane z tabeli. Zapytanie SELECT można również utworzyć w oknie Microsoft Query (omawianym w rozdziale 1.13. Wstawianie danych do tabeli, ich modyfikacja i usuwanie), które ułatwia wybranie tabel źródłowych, sposobu ich połączenia, wybrania kolumn czy odfiltrowania wierszy. Bazy są zwykle silnie znormalizowane, co skutkuje tym, że w tabeli często nie ma wszystkich potrzebnych nam danych, są one zastąpione przez kolumnę klucza obcego (zwykle z suffixem ID), która wskazuje na kolumnę klucza podstawowego w innej tabeli zwanej słownikową. Np. w tabeli oceny nie ma imienia, nazwiska czy klasy ucznia, a jedynie jego unikalny numer (ten sam, który wpisywany jest na świadectwie). Aby sprawdzić, który uczeń ma wstawioną ocenę należy odszukać imiona i nazwiska uczniów przypisane do ID umieszczonych w tabeli ocen. W Microsoft Query można również odfiltrować dane, np. wybrać z tabeli oceny tylko niedostateczne.

W celu ułatwienia pobierania danych utworzone zapytanie SELECT można zapamiętać w bazie danych tworząc **widok (View)**. Widok może połączyć dane z wielu tablic, odfiltrować je i pogrupować, co wymaga napisania często bardzo złożonego zapytania. Dzięki utrwaleniu go za pomocą widoku, zamiast powtórnie pisać skomplikowane zapytanie SELECT pobieramy wszystkie dane z widoku (oczywiście można je dodatkowo posortować, odfiltrować, pogrupować itd.). Do pobierania danych zamiast widoków można wykorzystać stworzone przez dba bardziej zaawansowane od widoków funkcje lub procedury składowane.

Czasami dane, potrzebne do wykonania raportu nie znajdują się na naszym serwerze. Znając adres zdalnego serwera, login i hasło można go podlinkować w swoim lokalnym serwerze. Pozwala to na pobieranie danych zewnętrznych (z bazy MsSQL, MySQL, Postgresql, bazy JET (w wersji 32 bit SQL) czyli Access, Excel, csv itd.) i łączenie ich w jednym zapytaniu z danymi lokalnymi i danymi pochodzącymi z różnych baz zewnętrznych, bez potrzeby ich ładowania i uaktualniania w lokalnej bazie. Ma to szczególne znaczenie w przypadku wersji Express, w której rozmiar bazy jest

ograniczony do 10GB. Korzystając w zapytaniu z wielu źródeł zewnętrznych można utworzyć w lokalnej bazie danych użytkownika widoki, które ułatwią pobieranie danych.

1.16. Pobieranie danych do Microsoft Excel

Jednym z bardziej znanych i lubianych programów jest Microsoft Excel. Do arkusza można dane wyeksportować, ale można również skopiować. Wynik zapytania SELECT (napisanego, odpalonego ze skryptu, zapakowanego w procedurę składowaną lub wyklikanego w Microsoft Query) jest domyślnie wyświetlany jako tabelka, którą można zaznaczyć, skopiować i wkleić do arkusza Microsoft Excel (lub dokumentu Microsoft Word, albo innego pliku). W przypadku arkusza Microsoft Excel należy jednak pamiętać, że w wersji 2003 obsługuje nieco ponad 64 tys. wierszy, od wersji 2007 około miliona, a w tabeli często jest znacznie więcej danych.

Microsoft Excel ma wbudowanego klienta SQL, dzięki czemu można bezpośrednio z niego wykonać zapytanie do bazy danych, a wynik umieścić w zwykłej tabeli lub tabeli przestawnej (albo wykresie i tabeli przestawnej) do dalszej analizy.

Do programu Microsoft Excel 2010 (i nowszych) jest dostępny dodatek o nazwie PowerPivot. Pozwala on na zgromadzenie pobranych z bazy i potrzebnych do analizy danych w tak zwanym indeksie x-velocity, co przyspiesza wykonywanie zaawansowanych raportów. Należy jednak pamiętać, że komputer powinien mieć odpowiednią ilość pamięci, ponieważ kilka kolumn pomnożonych przez kilkaset tysięcy wierszy potrzebuje nawet ponad 1GB RAM na utworzenie szybkiego indeksu x-velocity, przy danych z kilku dużych tablic może być potrzebne naprawdę dużo pamięci operacyjnej.

1.17. Zapytania do danych geograficznych (OPCJONALNE)

Microsoft SQL serwer w wersji 2008 lub wyższej obsługuje rysunkowe (spatial) typy danych. Rysunek lub plan w typie geometrycznym, a mapa w trybie geograficznym składa się ze **zbiorów punktów, linii łamanych i wieloboków**. Przy wykonaniu zapytania do takiej kolumny w oknie Query w oddzielnej zakładce tworzony jest odpowiedni rysunek. W przypadku typu geography jest on naniesiony na siatkę geograficzną. Do odpytywania danych geograficznych można użyć pakietu programów **SqlSpatialTools**. Program **SQLSpatial.exe** pozwala na bardziej efektywne wykonywanie zapytań z wykorzystaniem mapy, dodatkowo dane dotyczące np. trasy wycieczki wyświetlane są na tle odpowiedniego fragmentu konturowej mapy świata.

Jednym z popularnych **formatów map** są pliki **shp**. Jest dostępny bezpłatny program **mapbrowser**, który pozwala otwierać pliki shp. Do obszarów geograficznych w plikach shp może być przypisane wiele dodatkowych informacji np. nazwa, nazwa narodowa, kod regionu, liczba ludności itp. Z pliku shp definicje mapy można załadować do tabeli Microsoft SQL Serwer (w wersji 2008 lub nowszej) za pomocą programu **Shape2sql.exe** z pakietu SqlSpatialTools.

1.18. Wykonywanie raportów (OPCJONALNE)

Raporty (tabele, tabele przestawne Pivota, wykresy) z danych zgromadzonych w bazie można tworzyć za pomocą programu Microsoft Excel. Po pobraniu bezpłatnej wersji Microsoft SQL Serwer Express Edition with Advanced Feature można oprócz samego SQL zainstalować szereg usług towarzyszących, w tym usługi SQL Serwer Reporting Services (SSRS). W przypadku płatnej wersji standard lub enterprice dostępne są dodatkowe usługi, a w usłudze SSRS są dodatkowe funkcjonalności: ułatwiające tworzenie raportów, ich automatyczne wysyłanie na maila, przechowywanie wielu wersji raportu, mechanizmy pozwalające zwiększyć wydajność przy mniejszym obciążeniu serwera i bazy danych. Do tworzenia definicji raportów wykorzystuje się Visual Studio (narzędzie programistyczne) z szablonami BI (biznes inteligens, w tym RS). Po utworzeniu projektu reporting services można utworzyć raporty w postaci tabeli, macierzy (matrix), wykresu, wykresu na mapie (mapy od wersji SQL 2008R2).

2. Ćwiczenia praktyczne.

2.1. Ustawienie dostępu za pomocą protokołu TCPIP

- Uruchom program **SQL Server Configuration Manager**.
- Sprawdź w zakładce **SQL Server Services**, czy usługi pracują.
- Zmień ustawienia protokołów komunikacyjnych tak, aby umożliwić komunikację przez TCP/IP.
- Zrestartuj usługę **SQL Server (SQL2012Ex)**.

2.2. Nawiązywanie połączenia do serwera za pomocą programu SSMS

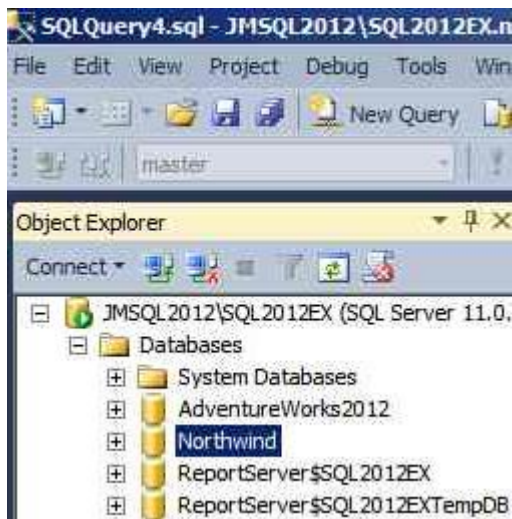
- Uruchom **Microsoft SQL Server Management Studio (SSMS)**.
- W oknie **Connect to Server** wskaż **Server name: .\SQL2012EX**, wybierz z listy **Authentication: Windows Authentication**.



- Podłącz się do serwera i przejrzyj listę obiektów w oknie **Object Explorer**.

2.3. Lista baz na serwerze

- Rozwiń listę serwerów w oknie **Object Explorer** programu **Microsoft SQL Server Management Studio (Administrator)**.
- Rozwiń listę obiektów instancji **SQL2012EX**.



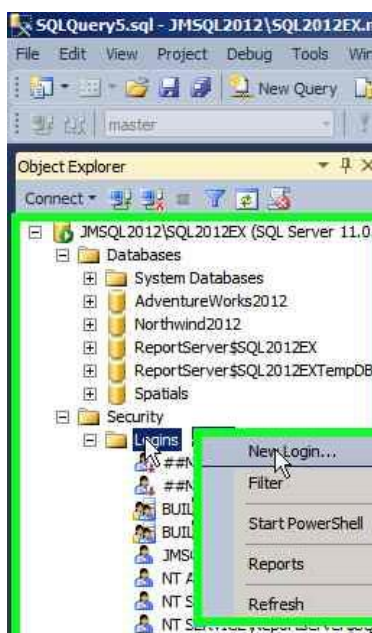
- Przejrzyj listę baz systemowych.

2.4. Loginy Windows i SQL

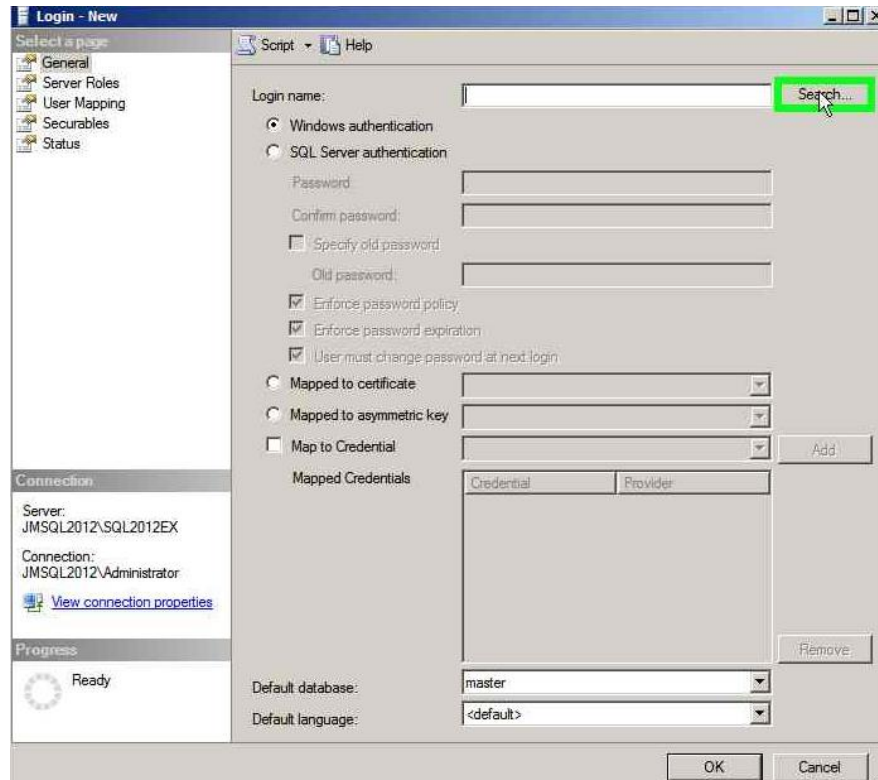
Utwórz loginy: użytkownika Windows, grupy Windows i login SQL.

a) Utwórz login użytkownika Windows o nazwie **Student**.

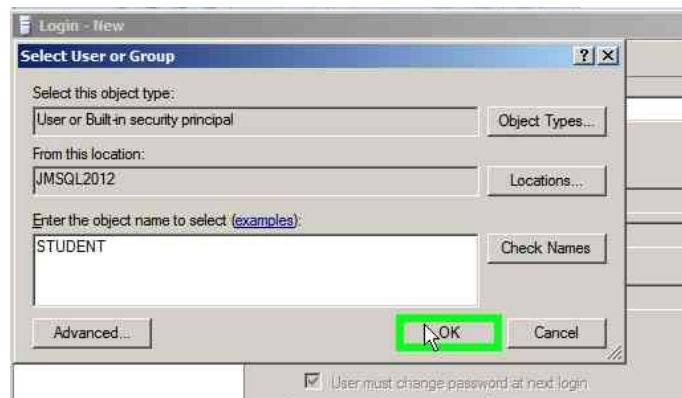
- Na liście obiektów instancji **SQL2012EX** odzyskaj obiekty **Logins**.



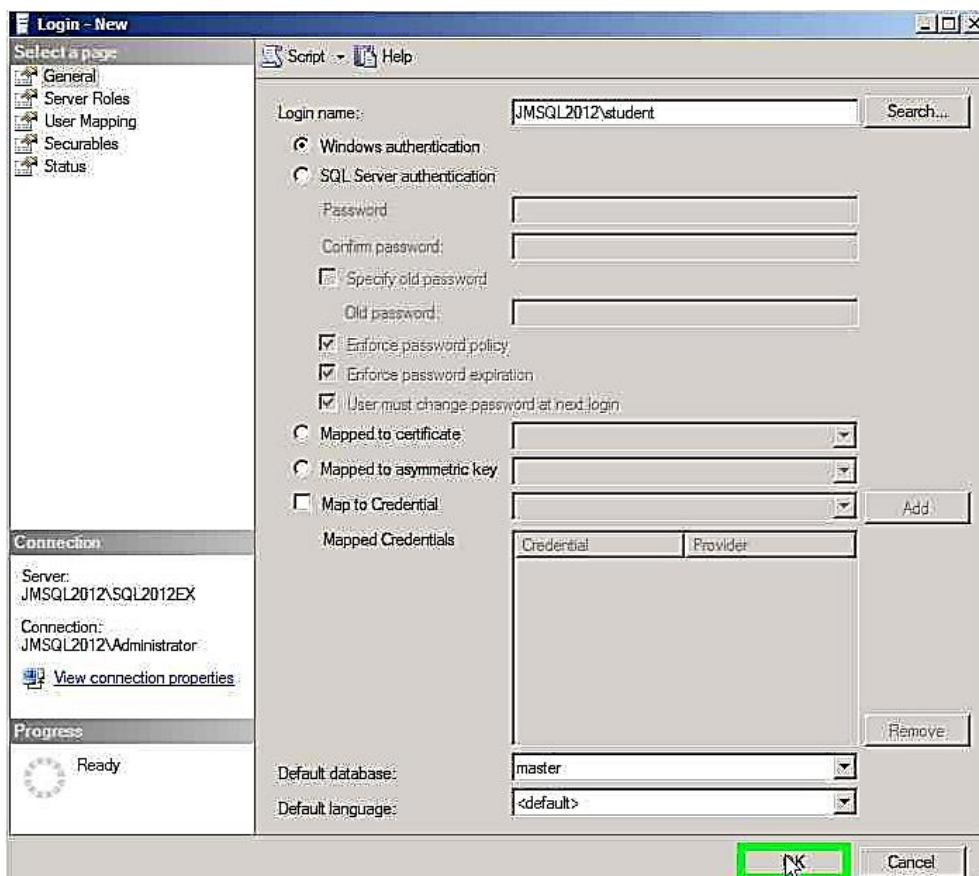
- Dodaj nowy login poleceniem **New Login...**



- Wyszukaj użytkownika o nazwie **Student**.

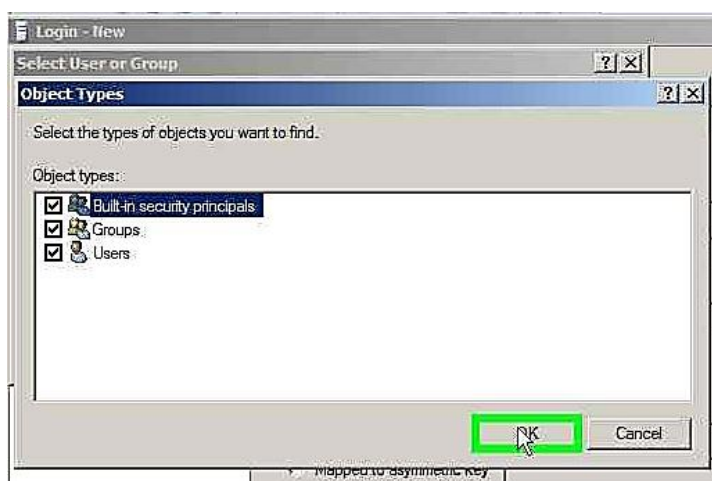


- Dodaj wskazanego użytkownika.



b) Utwórz login dla Grupy Windows.

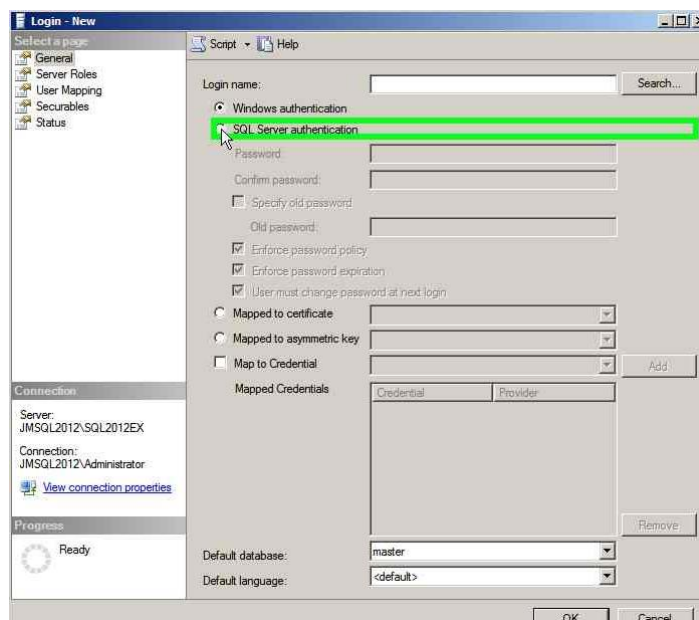
- Uruchamiaj polecenia analogiczne jak w przypadku dodawania użytkownika lokalnego systemu Windows.
- W oknie **Object Types** zaznacz flagę obok **Groups**.



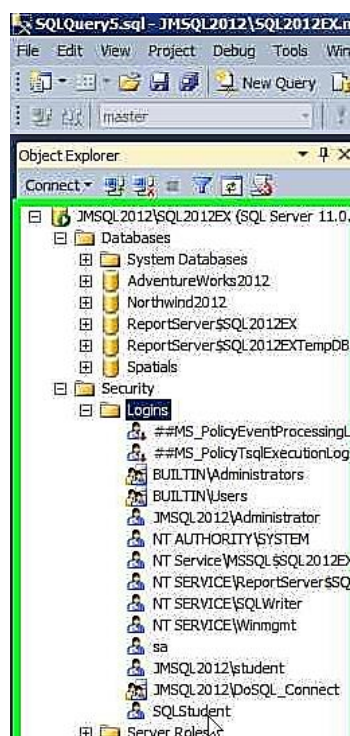
- W oknie **Select User or Group** wpisz w pole **Enter the object name to select:** DoSQL_Connect.
- Dodaj wskazany login.

c) Utwórz login SQL.

- Uruchamiaj polecenia analogiczne jak w przypadku dodawania użytkownika lokalnego systemu Windows.
- W oknie **Login - New** wskaż opcję **SQL Server authentication**, co spowoduje to aktywację opcji charakterystycznych dla loginów SQL.

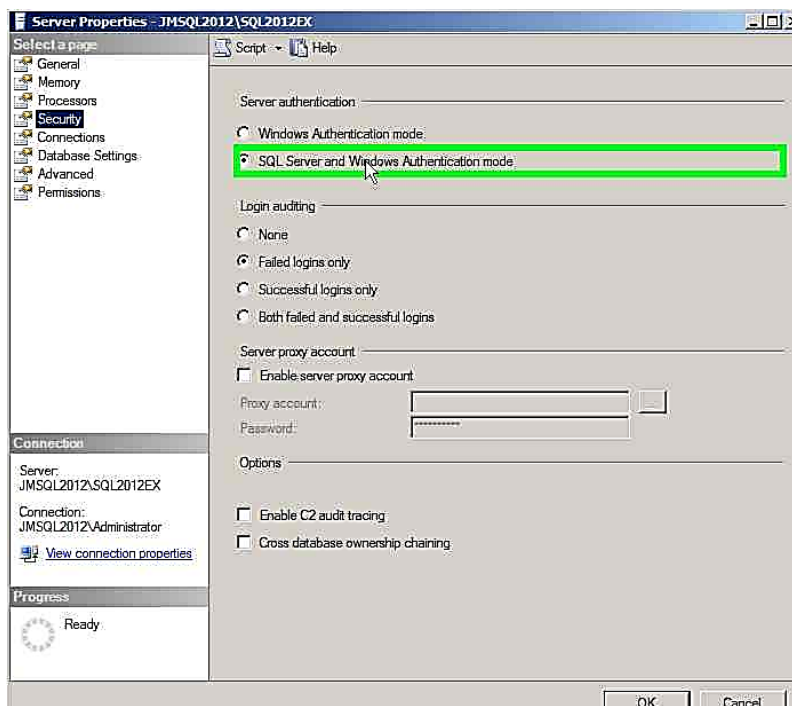


- Sprawdź czy są widoczne utworzone loginy w oknie **Object Explorer** w sekcji serwera **.\SQL2012EX, Security Logins**



- Aby było możliwe logowanie na konta SQL serwer musi mieć ustawiony tryb uwierzytelnienia MIX – zmień ustawienia **Security** w opcji **Properties** serwera

.\SQL2012EX

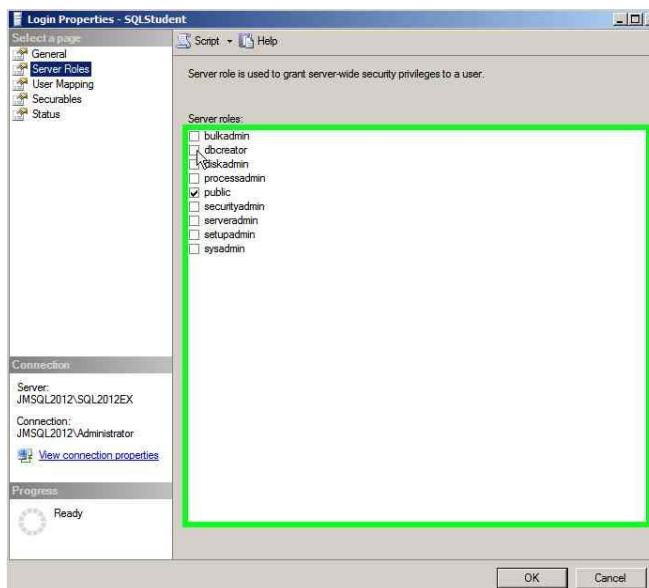


- Zmiana trybu security wymaga restartu usługi SQL Server :w oknie **Object Explorer**.

2.5. Uprawnienia nadawane za pomocą ról serwera

a) Nadaj loginowi **SQLStudent** uprawnienie tworzenia baz poprzez dodanie do roli **dbcreator**.

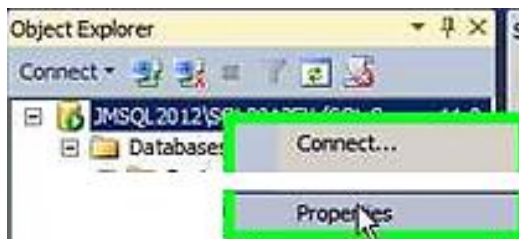
- Wskaż login **SQLStudent** w oknie **Object Explorer**.
- Zaznacz odpowiednią opcję w zakładce **Server Roles** okna z właściwościami konta.



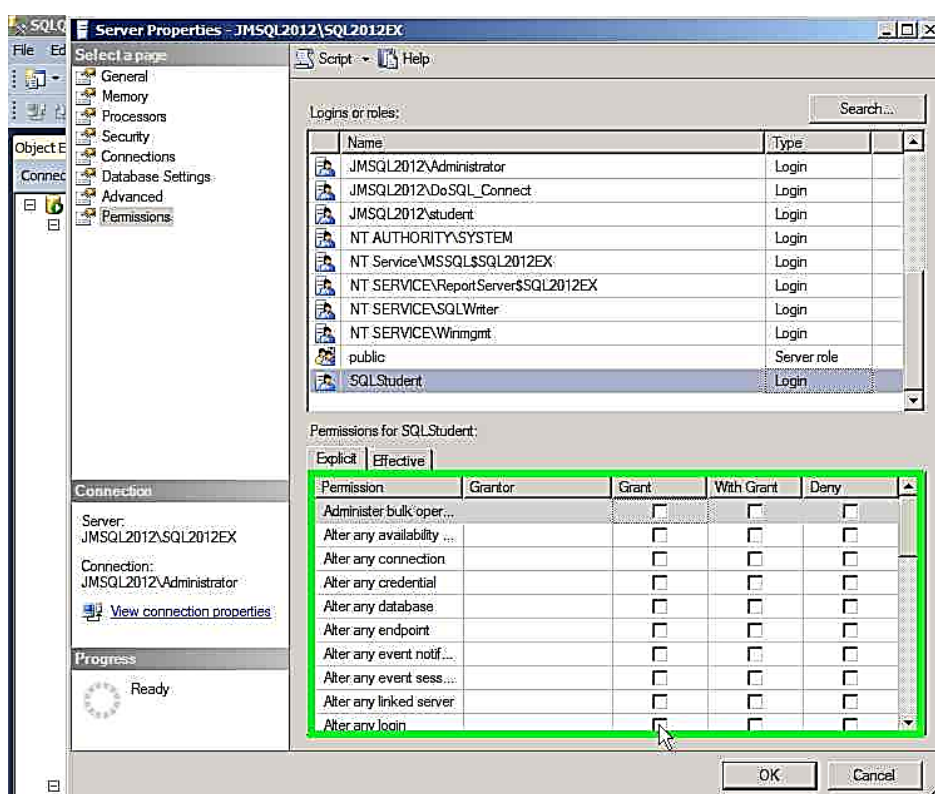
- Zaznacz opcję **dbcreator** i kliknij przycisk **OK**.

b) Nadaj uprawnienia bezpośrednio wybranemu loginowi

- Wyświetl okno właściwości serwera .\SQL2012EX



- W zakładce **Permissions** wybierz login **SQLStudent**.



- W sekcji **Explicit** można nadać unikalne uprawnienie.

Uprawnienia (kolumna **Permission**):

REVOKE - brak zdefiniowanych uprawnień, nie ma w wierszu uprawnienia flag

GRANT – dodatnie (flaga w kolumnie Grant) pozwala na wykonanie czynności

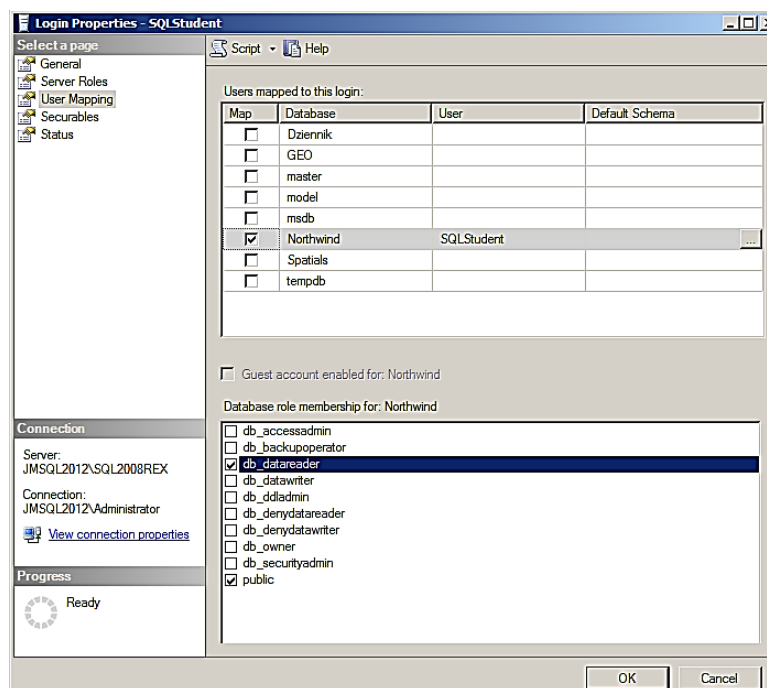
WITH GRANT – z prawem nadawania uprawnienia innym loginom (flagi w kolumnie Grant i With Grant)

DENY – zabronione (flaga w kolumnie Deny), np. użytkownik dodany do roli dbcreator i z zaznaczoną flagą w kolumnie Deny (wiersz Create Databases) nie może tworzyć baz.

2.6. Tworzenie użytkowników oraz role bazy danych i uprawnienia w bazie

Login może mieć przypisanego użytkownika w bazie danych. Dla loginu **SQLStudent** utwórz użytkownika w bazie **Northwind** i dodaj go do roli **db_datareader**.

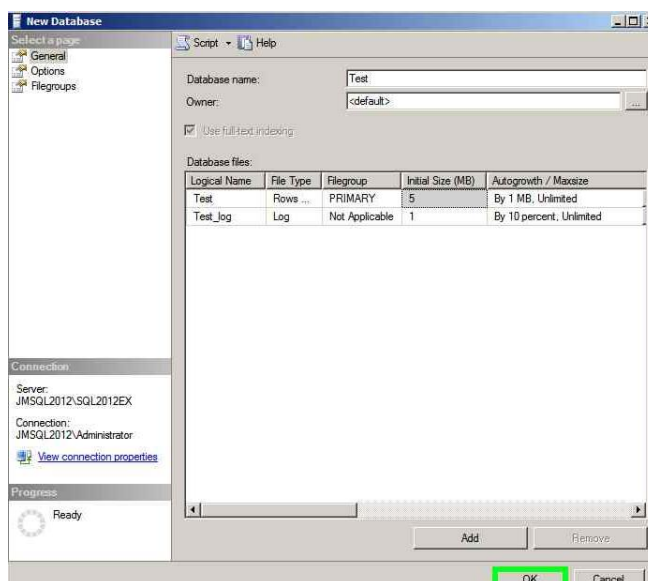
- We właściwościach loginu **SQLStudent** wybierz zakładkę **User Mapping**



- Wskaż bazę **Northwind** , wprowadź nazwę użytkownika i zaznacz flagę przy roli **db_datareader** w sekcji **Database role membership for Northwind**.

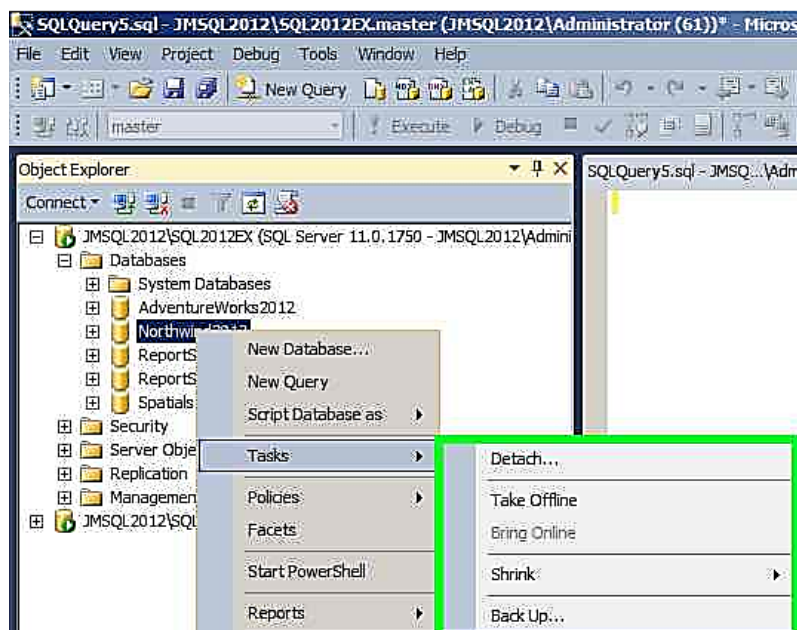
2.7. Tworzenie bazy danych za pomocą kreatora

Stwórz nową bazę o nazwie **Test**. Wykorzystaj polecenie **New Database...** z menu kontekstowego obiektu **Databases**.



2.8. Wykonywanie backupu pełnego bazy i odtwarzanie bazy z backupu

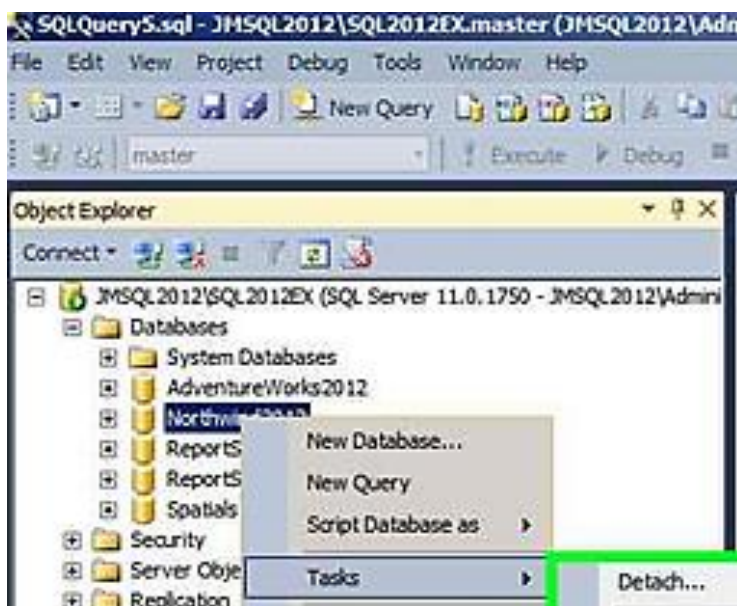
Wykonaj pełny backup bazy **Northwind**. Backup powinien być wykonany do pliku na dysku.



Nastąpiło uszkodzenie bazy **Northwind**. Odzyskaj ją z backupu nadpisując istniejącą bazę.

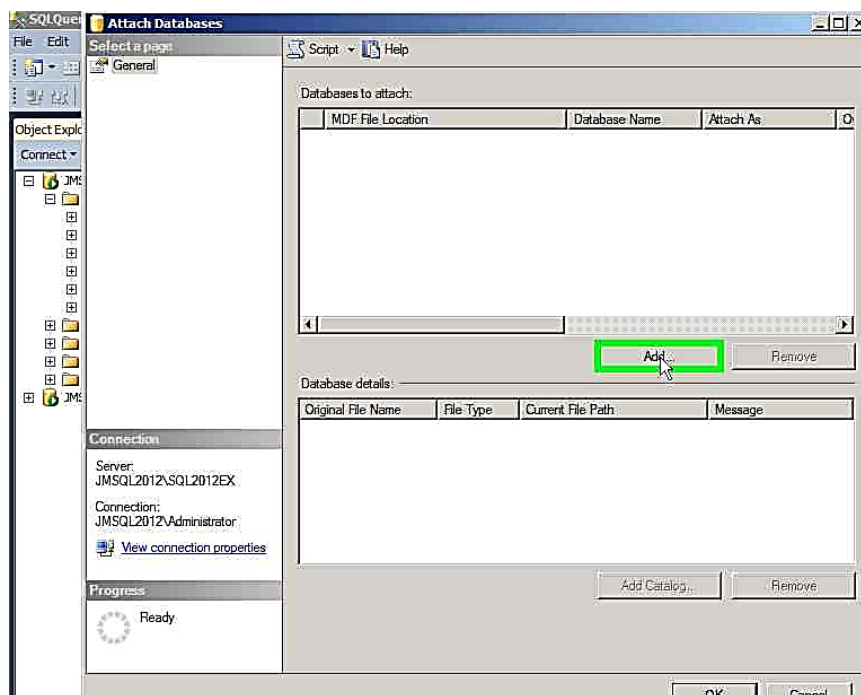
2.9. Montowanie bazy danych

Odmontuj bazę **Northwind**. Wykorzystaj polecenie **Detach Database**.

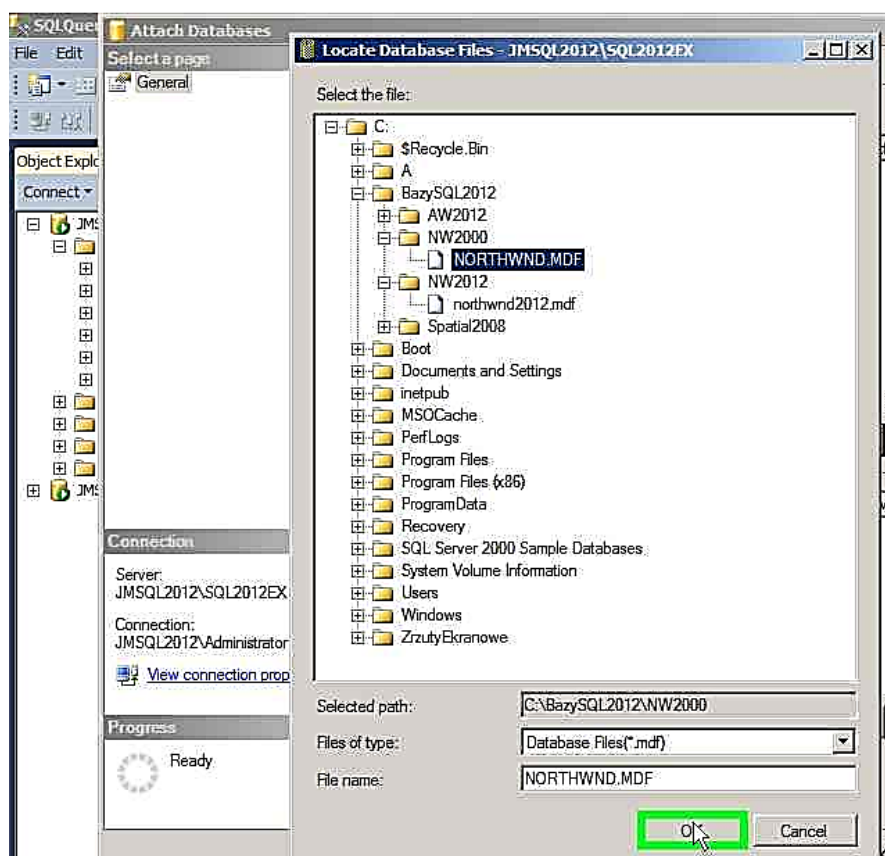


Po odmontowaniu można pliki bazy danych (plik mdf, ldf, mogą być również dodatkowe pliki ndf i ldf) skopiować do innej lokalizacji (do innego folderu, na inny dysk, na inny serwer ...)

Zamontuj bazę **Northwind**. Odpowiednie polecenie (**Attach...**) znajduje się w menu kontekstowym obiektu **Databases**.

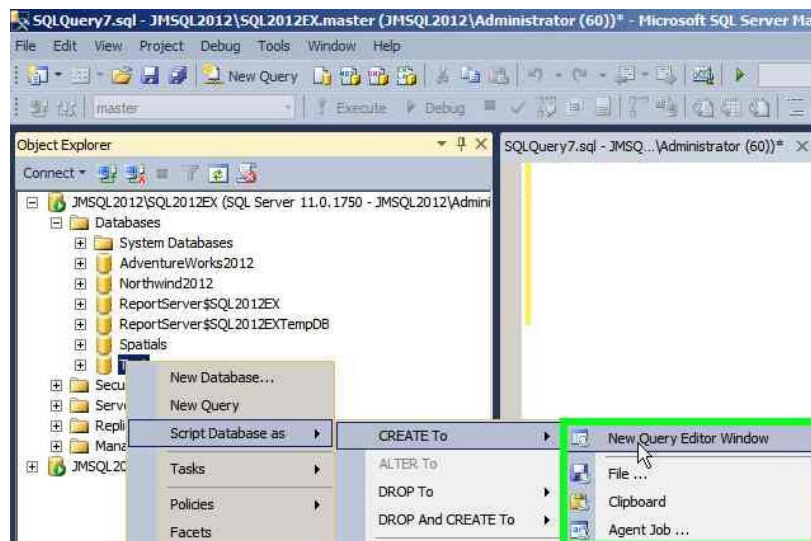


Należy wskazać lokalizację plików bazy danych.



2.10. Tworzenie bazy za pomocą skryptu

Po utworzeniu obiektu np. bazy można wygenerować skrypty SQL. Wygeneruj skrypt do bazy **Test**. Utworzony skrypt można uruchomić na innym serwerze.



Utwórz bazę Dziennik wykorzystując skrypt **dziennik.sql** z folderu **c:\Nauczyciele**. Po otwarciu skryptu w SSMS uruchom go wykorzystując przycisk **! Execute**.

Zawartość pliku Dziennik.sql :

```
USE master
```

```
GO
```

```
-- DROP DATABASE Dziennik;
```

```
CREATE DATABASE Dziennik;
```

```
GO
```

```
USE Dziennik;
```

```
GO
```

```
CREATE TABLE Przedmioty (
```

```
    PrzedmiotID int Identity Primary Key
```

```
, PrzedmiotNazwa nvarchar(50) NOT NULL
```

```
, Opis XML
```

```
);
```

```
CREATE TABLE Nauczyciele (
```

```
    NauczycielID int Identity Primary Key
```

```
, Nazwisko nvarchar(50) NOT NULL
```

```
, Imie nvarchar(50) NOT NULL
, Telefon char(9)
, Mail nvarchar(255)
, opis XML
);
```

```
CREATE TABLE Rodzice (
  RodzicID int Identity Primary Key
, Nazwisko nvarchar(50)
, Imie nvarchar(50)
, Telefon char(9)
, Mail nvarchar(255)
, opis XML
);
```

```
CREATE TABLE Klasy (
  KlasaID int Identity Primary Key
, KlasaNazwa char(2) NOT NULL
, RokSzkolny char(9) NOT NULL
, DataOd smalldatetime
, DataDo smalldatetime
, Wychowawca int NOT NULL
, Opis XML
, CONSTRAINT FK_Wychowawca
  FOREIGN KEY (Wychowawca) REFERENCES Nauczyciele (NauczycielID)
);
```

```
CREATE TABLE Uczniowie (
  UcenID int Primary Key --Numer na świadectwie
, DziennikNr tinyint NOT NULL
, KlasaID int NOT NULL
, Nazwisko nvarchar(50)
, Imie nvarchar(50) NOT NULL
, Ojciec int NULL
, Matka int NULL
, DataUr smalldatetime
, Uwagi XML
, CONSTRAINT FK_UKlasaID
  FOREIGN KEY (KlasaID) REFERENCES Klasy (KlasaID)
, CONSTRAINT FK_Ojciec
```

```
FOREIGN KEY (Ojciec) REFERENCES Rodzice (RodzicID)
, CONSTRAINT FK_Matka
FOREIGN KEY (Matka) REFERENCES Rodzice (RodzicID)
);

CREATE TABLE PlanLekcji (
    PlanLekcjiID int Identity Primary Key
, KlasaID int
, NauczycielID int
, PrzedmiotID int
, GodzinySale XML
, CONSTRAINT FK_PLKlasaID
FOREIGN KEY (KlasaID) REFERENCES Klasy (KlasaID)
, CONSTRAINT FK_PLNauczycielID
FOREIGN KEY (NauczycielID) REFERENCES Nauczyciele (NauczycielID)
, CONSTRAINT FK_PLPrzedmiotID
FOREIGN KEY (PrzedmiotID) REFERENCES Przedmioty (PrzedmiotID)
);

CREATE TABLE OcenyZa (
    OcenyZaID Tinyint Primary Key
, OcenaZaOpis nvarchar(50)
);

INSERT INTO OcenyZa (OcenyZaID, OcenaZaOpis)
VALUES
    (0, 'Inne')
, (1, 'Praca klasowa')
, (2, 'Kartkówka')
, (3, 'Odpowiedź ustna')
, (4, 'Praca domowa')
, (5, 'Praca na lekcji')
, (6, 'Ćwiczenia')
, (7, 'Aktywność')
, (8, 'Konkurs')
, (9, 'Praca w grupach')
, (10, 'Projekt')
GO

CREATE TABLE Oceny (
```

```
OcenaID int Identity Primary Key
, UcenID int NOT NULL
, PlanLekcjiID int NOT NULL
, Data smalldatetime DEFAULT getdate() NOT NULL
, Ocena tinyint NOT NULL
, OcenyZaID Tinyint NOT NULL DEFAULT 0
, Opis nvarchar(max)
, Opis2 XML
, CONSTRAINT FK_OcenyUczenID
  FOREIGN KEY (UczenID) REFERENCES Uczniowie (UczenID)
, CONSTRAINT FK_OcenyPlanLekcjiID
  FOREIGN KEY (PlanLekcjiID) REFERENCES PlanLekcji (PlanLekcjiID)
, CONSTRAINT FK_OcenyOcenyZaID
  FOREIGN KEY (OcenyZaID) REFERENCES OcenyZa (OcenyZaID)
);
GO
```

```
CREATE RULE R_Oceny
AS
    @value IN (0,1,2,3,5,6)
GO
```

```
EXEC sp_bindrule R_Oceny, 'Oceny.Ocena'
GO
```

```
CREATE TABLE Frekwencja (
  FrekwencjaID int Identity Primary Key
, Data smalldatetime
, GodzinLekcyjnych tinyint
, GodzinNieobecnych tinyint
, WTymNieusprawiedliwionych tinyint
, Spoznien tinyint
, Uwagi nvarchar(200)
, Uwagi2 XML
);
```

```
CREATE TABLE UwagiPochwaly (
  UwagaID int Identity Primary Key
, UcenID int NOT NULL
, NauczycielID int NOT NULL
```

```
, Data smalldatetime NOT NULL
, Pozytywna Bit NOT NULL DEFAULT 0
, Uwaga nvarchar(max)
, CONSTRAINT FK_UwagiUczenID
  FOREIGN KEY (UczenID) REFERENCES Uczniowie (UczenID)
, CONSTRAINT FK_UwagiNauczycielID
  FOREIGN KEY (NauczycielID) REFERENCES Nauczyciele (NauczycielID)
);
```

```
CREATE TABLE OcenySemestralne (
  OcenySemestralneID int Identity Primary Key
, Semestr tinyint DEFAULT 1 NOT NULL
, KlasaID int NOT NULL
, UczenID int NOT NULL
, PrzedmiotID int NOT NULL
, Ocena tinyint NOT NULL
, Data smalldatetime
, Uwagi XML
, CONSTRAINT FK_OcenySemKlasaID
  FOREIGN KEY (KlasaID) REFERENCES Klasy (KlasaID)
, CONSTRAINT FK_OcenySemUczenID
  FOREIGN KEY (UczenID) REFERENCES Uczniowie (UczenID)
, CONSTRAINT FK_OcenySemPrzedmiotID
  FOREIGN KEY (PrzedmiotID) REFERENCES Przedmioty (PrzedmiotID)
)
GO
```

```
EXEC sp_bindrule R_Oceny, 'OcenySemestralne.Ocena'
```

```
CREATE TABLE Ankiety (
  AnkietaID int Identity Primary Key
, AnkietaNazwa nvarchar(50) NOT NULL
, AnkietaPytania XML
);
```

```
CREATE TABLE AnkietyWyp (
  AnkietaWypID int Identity Primary Key
, AnkietaID int NOT NULL
, Wynik tinyint NOT NULL
, Opis XML
```

, Odpowiedzi XML
);

2.11. Typy danych

Przedyskutujcie w grupie wybór odpowiedniego typu danych.

2.12. Tworzenie tabel

Utwórz w bazie **Test** tabelę o nazwie **Szkolenie**. Domyślnie zapisywanie utworzonych w interfejsie graficznym tabel jest zablokowane. Aby odblokować tę funkcjonalność w SSMS wybierz z menu **Tools > Options**, menu **Designers**, zaznacz **Table and Database Designers** i odznacz flagę przy opcji **Prevent saving changes that require table re-creation**.

Kolumny tablicy **Szkolenie** powinny być następujące:

SzkolenieID int

SzkolenieNazwa nvarchar(50)

SzkolenieOpis XML

Kolumna **SzkolenieOpis** powinna zezwalać na zapis wartość null. Kolumna **SzkolenieID int** powinna być kluczem głównym tablicy, a wartości powinny być do niej automatycznie wstawiane jako kolejne wartości (Identity, Identity Increment =1). Kolumna **SzkolenieNazwa** powinna pozwalać na zapis polskich znaków (**Collation** wybierz **SQL_Polish_CP1250_CS_AS**).

2.13. Wstawianie danych do tabeli, ich modyfikacja i usuwanie

Dodaj do tabeli **Szkolenie** kilka szkoleń (polecenie z menu kontekstowego **Edit Top 200 Rows**). Wpisz w kolejnych wierszach: **Administracja SQL 2012**, **Administracja Windows 2012**, **Office 365**, **Błąd**, **Wdrażanie Microsoft SQL 2012**. Następnie popraw w 3 wierszu **Office 365** na **Microsoft Office 365**. i usuń wiersz 4 z nazwą szkolenia **Błąd**.

2.14. Importowanie danych z tabeli Microsoft Excel

a) Zaimportuj dane z plików Microsoft Excel do tabel Microsoft SQL.

W folderze **c:\nauczyciele\PlikiDoBazyDziennik** znajdują się trzy pliki: Microsoft Excel: **Nauczyciele.xlsx**, **Klasy.xlsx**, **Nazwiska_Imiona.xlsx**.

W pliku **Nauczyciele.xlsx** w arkuszu **Kadra** zostały umieszczone w czterech kolumnach: **Nazwisko**, **Imie**, **Telefon**, **Mail**. Zakres komórek **\$a\$1:\$d\$23** z informacjami

o nauczycielach jest tabelą o nazwie **Nauczyciele**.

W pliku **Klasy.xlsx** znajdują się dwa arkusze z danymi: Klasy i Nauczyciele. W arkuszu **Klasy** dane są w 4 kolumnach: **KlasaNazwa**, **RokSzkolny**, **Wychowawca**, **NazwiskoNaucz**. W kolumnie **NazwiskoNaucz** jest dynamiczna lista wyboru zawierająca nazwiska nauczycieli załadowane do bazy (zamiast nazwiska można było wybrać zapytanie `[Nazwisko]+' '+[Imie]`)

	A	B	C	D	E
1	KlasaNazwa	RokSzkolny	Wychowawca	NazwiskoNaucz	
2	IA	2012/2013		19 Kowalski	
3	IB	2012/2014		4 Kowalczyk	
4	IC	2012/2015		6 Lewandowska	
5	IIA	2012/2016		7 Zielińska	
6	IIB	2012/2017		5 Kamińska	
7	IIC	2012/2018		10 Kamińska	
8	IIIA	2012/2019		11 Lewandowska	
9	IIIB	2012/2020		18 Zielińska	
10	IIIC	2012/2021		20 Szymborska	
11	IVA	2012/2022		8 Wozniak	
12	IVB	2012/2023		3 Dąbrowska	
13	IVC	2012/2024		8 Kozłowska	
				3 Jankowska	
				7 Wójcik	
				7 Zielińska	

Lista rozwijana została utworzona za pomocą reguły poprawności

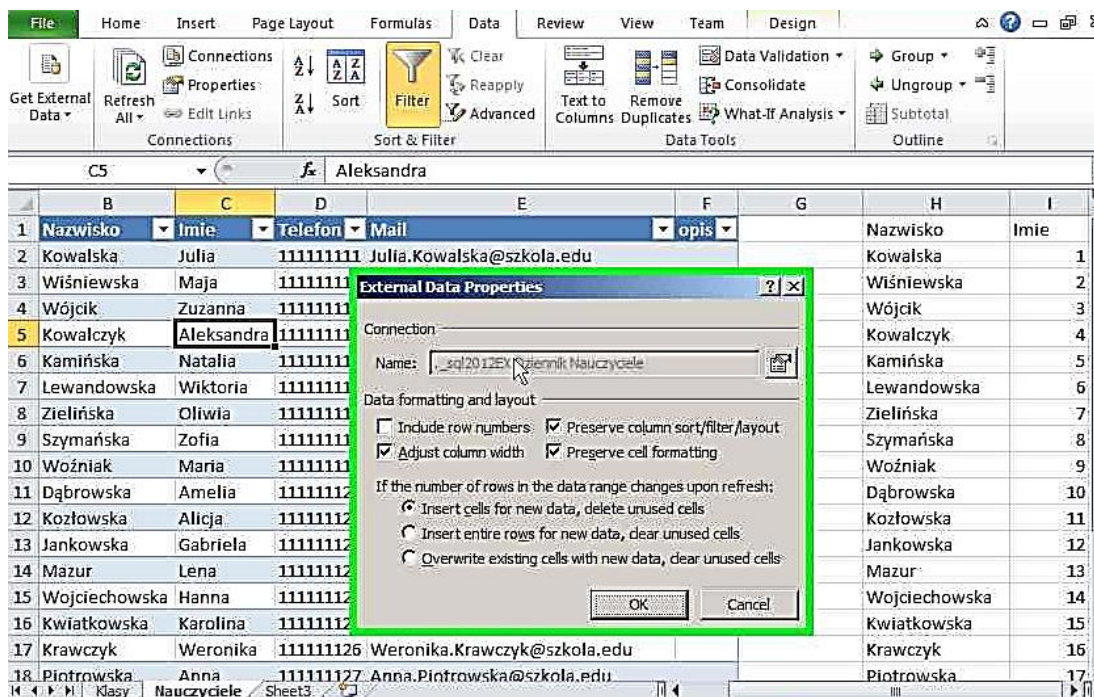
	A	B	C	D	E	F	G	H
4	IC	2012/2015		6 Lewandowska				
5	IIA	2012/2016						
6	IIB	2012/2017						
7	IIC	2012/2018						
8	IIIA	2012/2019						
9	IIIB	2012/2020						
10	IIIC	2012/2021						
11	IVA	2012/2022						
12	IVB	2012/2023						
13	IVC	2012/2024						
14								
15								
16								
17								
18								
19								
20								

Data Validation dialog box settings:

- Validation criteria: Allow: List
- Source: `=Nauczyciele!B2:B23`
- Ignore blank:
- In-cell dropdown:

Lista pobiera wartości z arkusza **Nauczyciele** z tabeli o adresie: **\$B\$2:\$B\$23**

W kolumnie **Wychowawca** wartość jest obliczana poprzez znalezienie w tabeli **Nauczyciele!\$H\$1:\$I\$24** (arkusz **Nauczyciele**, tabela **\$H\$1:\$I\$24**) wybranego nazwiska nauczyciela i pobranie z drugiej kolumny identyfikatora nauczyciela). Do arkusza **Nauczyciele** dane są pobierane z serwera **.\\SQL2012Ex**, bazy **Dziennik**, tabeli **Nauczyciele**.



Po każdym wstawieniu do tabeli sql nowych nauczycieli należy dane odświeżyć



Pobieranie danych z bazy SQL do arkusza Microsoft Excel jest omówione w rozdziale 1.6, a ćwiczenia zostaną wykonane w rozdziale 2.16.

- b) Pobierz dane z pliku **Nauczyciele.xlsx** do tabeli **Nauczyciele** w bazie **Dziennik** za pomocą kreatora **SQL Server Import and Export** programu SSMS.

2.15. Pobieranie danych

- a) Pobierz dane z tabeli **dbo.Szkolenie** w bazie **Test**. Skorzystaj ze wskazanych metod:
 - Wybierz polecenie **Select Top 1000 Rows...** w menu kontekstowym obiektu tablicy **dbo.Szkolenie**. Zostanie wygenerowane i automatycznie wykonane zapytanie SQL. Wynik w zakładce **Result** można zaznaczyć, skopiować (lub skopiować z nagłówkami) i wkleić do innego programu np. arkusza Microsoft Excel lub dokumentu Microsoft Word.

The screenshot shows the SQL Server Enterprise Manager interface. The top toolbar includes an 'Execute' button. The main window displays a query script for 'SQLQuery3.sql - JMSQL2012\...'. The script is as follows:

```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP 1000 [SzkolenieID]
      , [SzkolenieNazwa]
      , [SzkolenieOpis]
FROM [Test].[dbo].[Szkolenie]

```

Below the script, the 'Results' tab shows the following data:

	SzkolenieID	SzkolenieNazwa	SzkolenieOpis
1	1	Administracja SQL 2012	NULL
2	2	Administracja Windows 2012	NULL
3	3	Microsoft Office 365	NULL
4	5	Wdrażania Microsoft SQL 2012	NULL

- Z menu kontekstowego tablicy **dbo.Szkolenie** wybierz **Script Tables as > SELECT To > New Query Editor Window**. Wygenerowany zostanie skrypt z zapytaniem SELECT. Aby go wykonać wciśnij przycisk **! Execute**.
- Po wyborze polecenia **Edit Top 200 Rows** z menu kontekstowego tablicy **dbo.Szkolenie**, zmień typ polecenia (menu **Change Type** - wybierz **Select**). Włącz wyświetlanie tablic, narzędzia kryteriów filtrowania i kodu sql wciskając przyciski



na lewo od listy **Change Type**. W oknie edycji zapytania możesz zmieniać zapytanie, np. dodać kryteria wyszukiwania - w wierszu **SzkolenieID** w kolumnie **Filter** wpisz **>2**. Zmienione zapytanie można wykonać klikając przycisk **! Execute**.

The screenshot shows the 'Change Type' dialog box for the 'Szkolenie' table. The 'Filter' column in the results grid is highlighted with the value '> 2'. The SQL query is updated as follows:

```

SELECT TOP (200) SzkolenieID, SzkolenieNazwa, SzkolenieOpis
FROM Szkolenie
WHERE (SzkolenieID > 2)

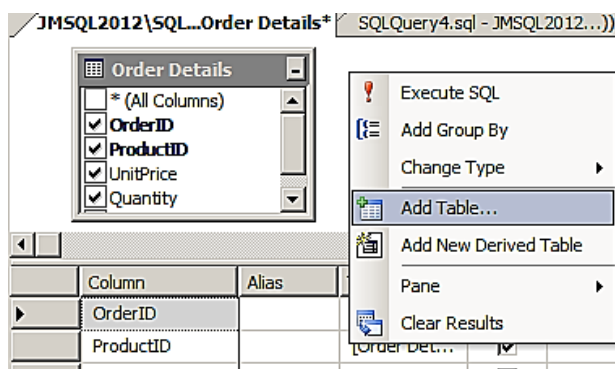
```

The results grid now shows only the rows where the ID is greater than 2:

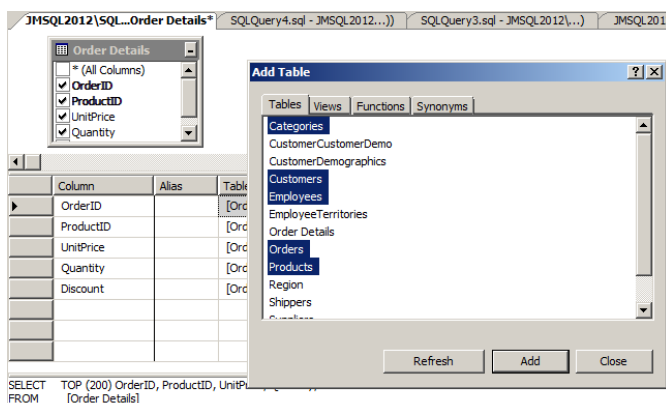
	SzkolenieID	SzkolenieNazwa	SzkolenieOpis
▶	3	Microsoft Office ...	NULL
	5	Wdrażania Micro...	NULL
*	NULL	NULL	NULL

b) Pobierz dane z tabeli **Northwind.dbo.[Order Details]**, zamiast kolumn z identyfikatorem klucza obcego wyświetl potrzebne informacje z odpowiednich tabel.

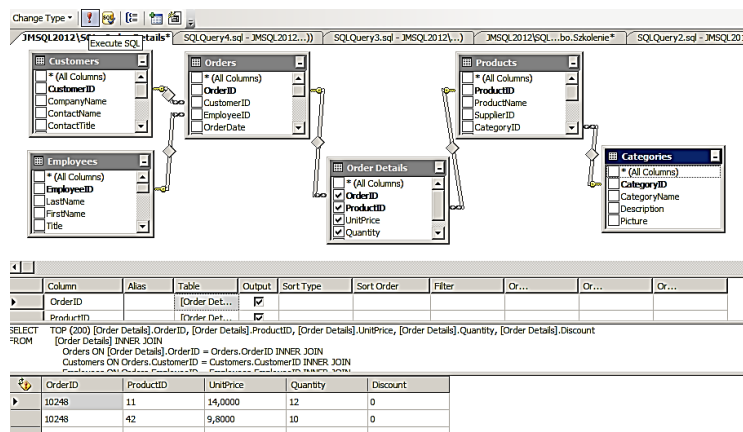
- W oknie edycji zapytania dodaj potrzebne tablice za pomocą polecenia **Add Table...**



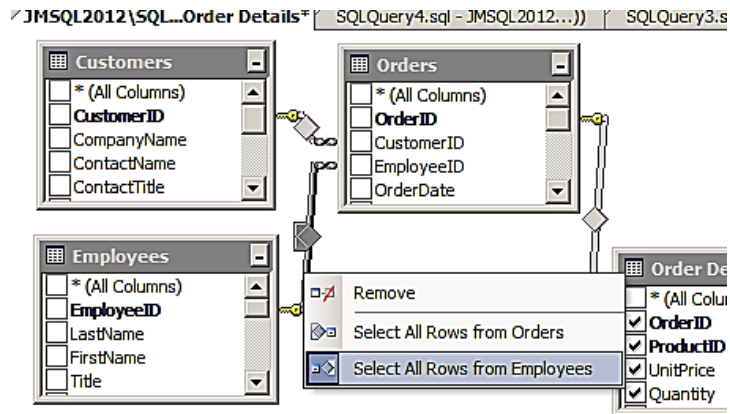
- Dodaj tablice Categories, Customers, Employees, Orders, Products.



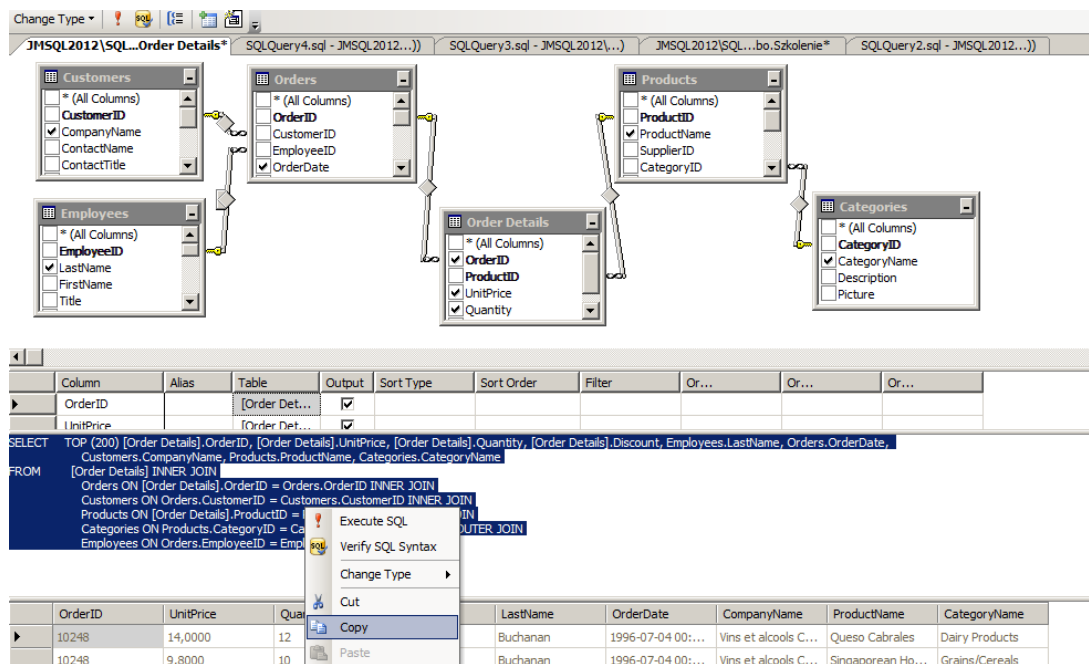
- W oknie edycji wyświetlone zostaną dodane tabele i powiązania między nimi. Domyślnie pobierane są tylko te wiersze, dla których kolumny kluczy łączących tabele są takie same.



- Np. pracownik, który nic nie sprzedał nie zostanie wyświetlony. Zmień powiązania pomiędzy tablicą **Orders** i **Employees** na **Select All Rows from Employees**.



- W oknie edycji zaznacz kolumny: **Employees.LastName**, **Customers.CompanyName**, **Orders.OrderDate**, **[Order Details].OrderID**, **[Order Details].UnitPrice**, **[Order Details].Quantity**, **Products.ProductName**, **Categories.CategoryName** jak na rysunku poniżej.

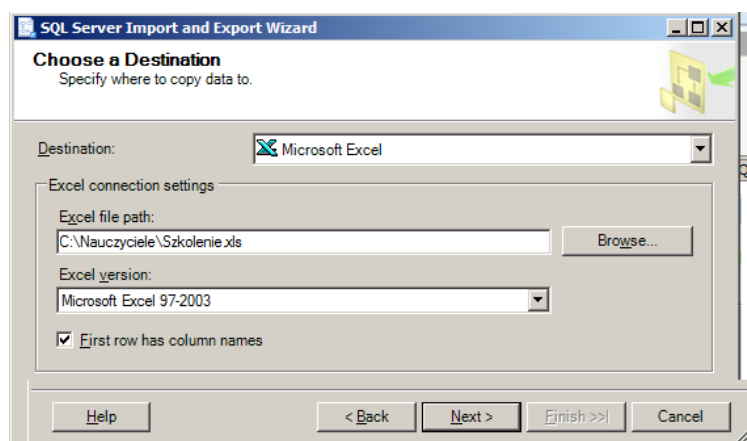
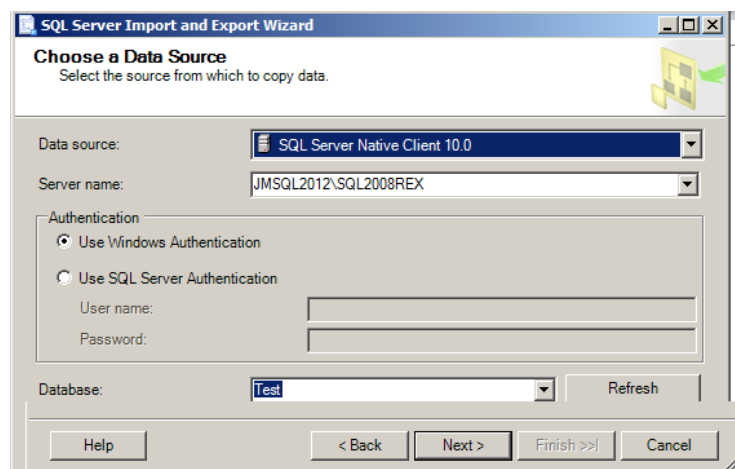


- Wciśnij przycisk **! Execute SQL**.
- c) Utwórz zapytanie i zapisz je na serwerze jako widok. Wykorzystaj polecenie **New View...** z menu kontekstowego obiektu **Views**. Stwórz widok łączący tabele **Categories**, **Customers**, **Employees**, **Orders**, **Products**.

2.16. Pobieranie danych do Microsoft Excel

- a) Wyeksportuj dane z tabeli **Test.dbo.Szkolenie** do pliku Microsoft Excel xls. Wykorzystaj narzędzie **SQL Server Import and Export Wizard**. Jako źródło danych wskaż **SQL Server Native Client 10.0**, dane mają być eksportowane do **Microsoft Excel**. Wyeksportowane

dane są statyczne, czyli zmiana danych w tabeli sql nie ma wpływu na zawartość arkusz Excel.



b) W programie Microsoft Excel, w nowym arkuszu utwórz połączenie do bazy **Test** i pobierz dane z tabeli **Szkolenie**.

- Powinieneś wybrać zakładkę **Data (Dane** w wersji polskiej MS Excel) w nowym skoroszycie Microsoft Excel, a następnie z menu **From Other Sources (Z innych źródeł** w wersji polskiej) wybrać polecenie **From SQL Server**. W parametrach połączenia podaj:

Server name: wpisz **.\SQL2012EX**,

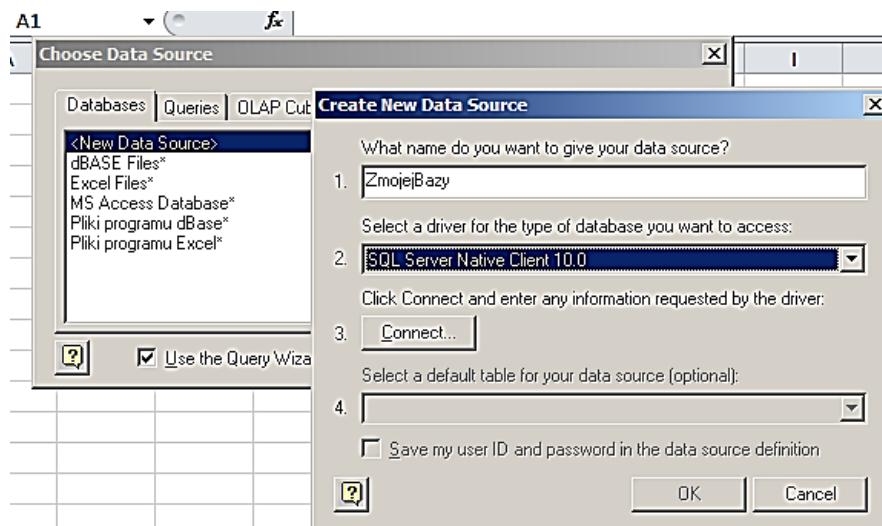
Log on credentials pozostaw domyślny wybór **Use Windows Authentication**.

- Do arkusza Excel zostaną pobrane dane z tabeli **Test.dbo.Szkolenie**. Dane są zapisywane w arkuszu, ale przy otwieraniu pliku nawiązywane jest połączenie z bazą w celu aktualizacji danych. Można też uaktualnić dane wciskając przycisk **Refresh All**

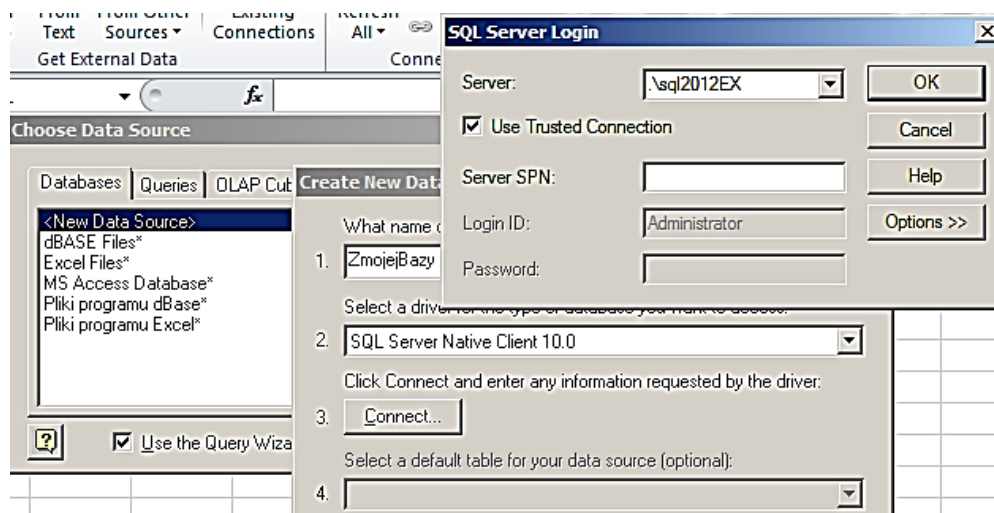


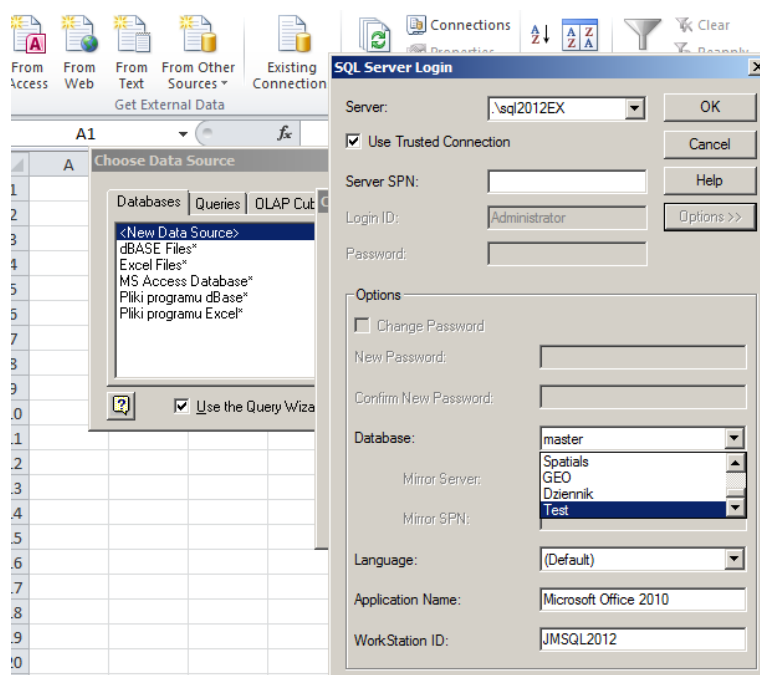
- c) Do pobrania i odfiltrowania danych można również użyć Microsoft Query, w technologii dostępnej w Microsoft Excel 2003 i starszych. Pobierz dane do MS Excel wskazując jako źródło Microsoft Query.

- Zdefiniuj nowe źródło danych o nazwie **ZmojejBazy**, jako sterownik wskaż **SQL Server Native Client 10.0**.

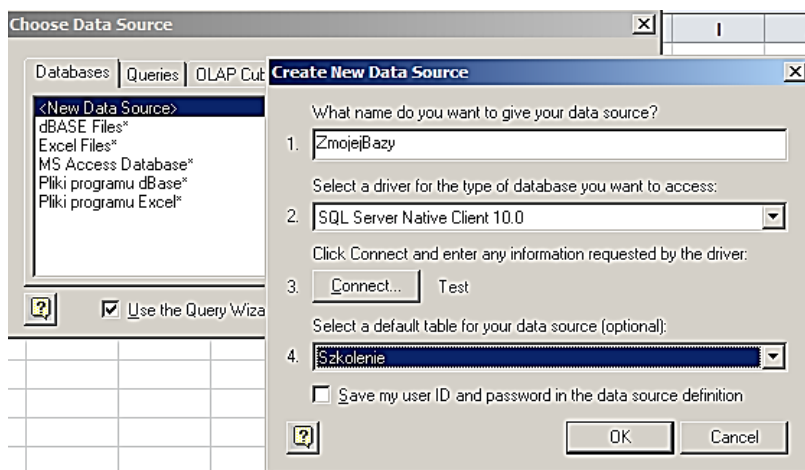


- W parametrach połączenia wskaż serwer **.\SQL2012EX**, zaznacz flagę opcji **Use Trusted Connection**.

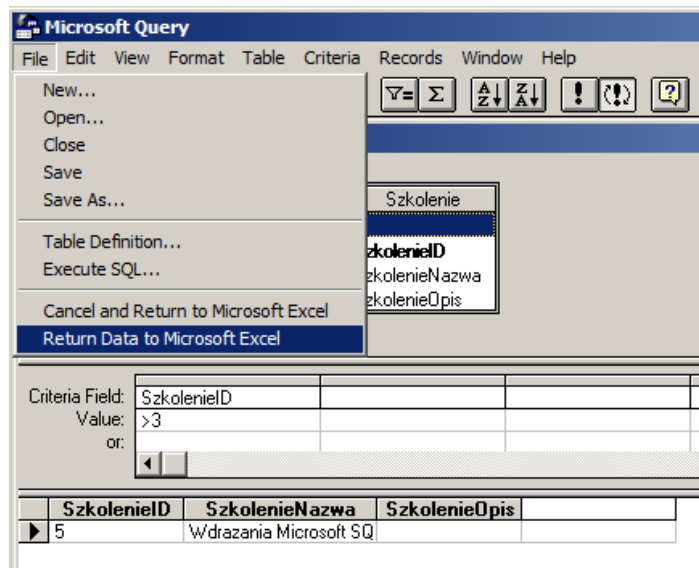




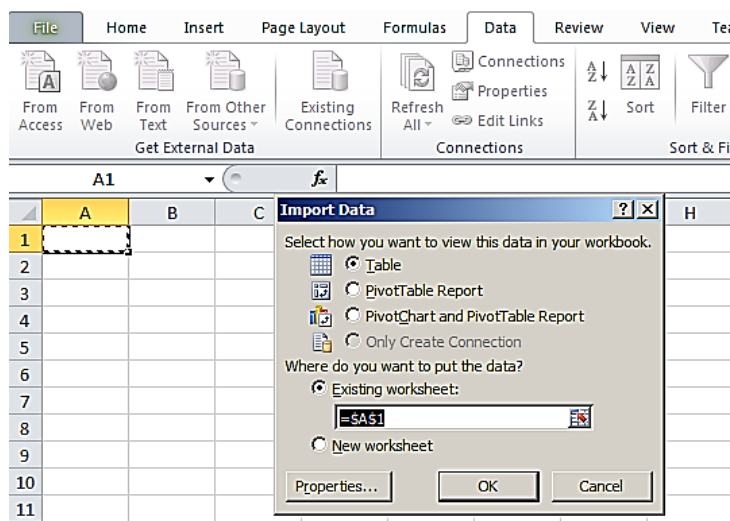
- Po zdefiniowaniu nowego źródła danych o nazwie **ZmojeJBazy** wykorzystaj je do podłączenia do bazy **Szkolenie**.



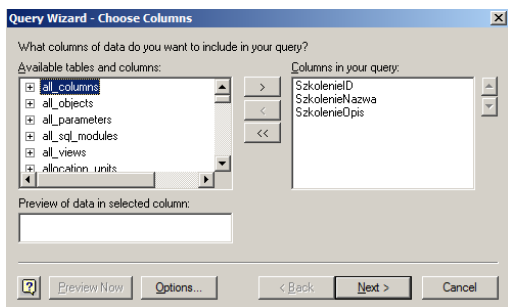
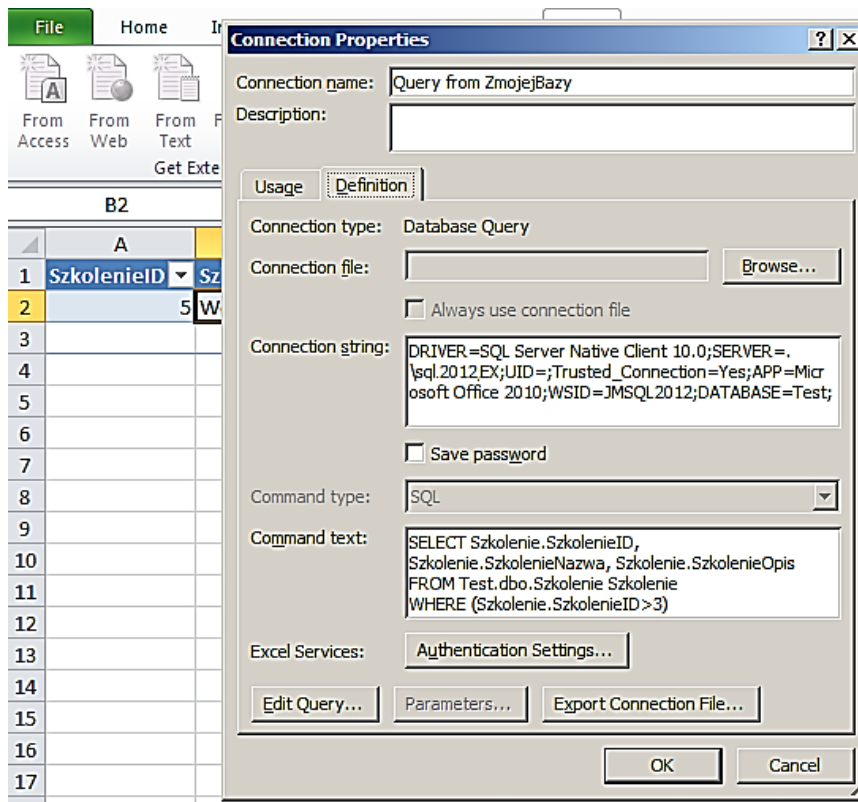
- W oknie Microsoft Query możesz dodawać tabele, wybierać kolumny, filtrować dane analogicznie jak w oknie dostępnym w SSMS poleceniem Edit Top 200 Rows.



- W oknie **Import Data** wskaż sposób prezentacji pobranych danych w Excelu.



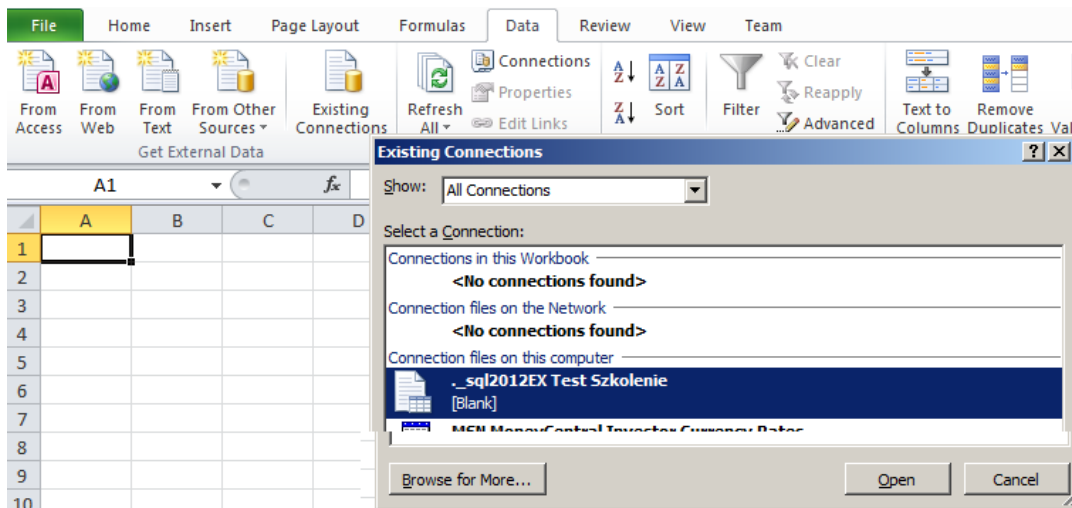
- W arkuszu Microsoft Excel znajdą się dane z tabeli zgodnie z utworzonym zapytaniem. Można zmienić zapytanie wybierając z listy **Refresh All** opcję **Connection Properties...** i edytując zapytanie po wybraniu przycisku **Edit Query**.



W oknie **Query Wizard – Choose Columns** można wybrać inną tabelę i w kolejnych etapach kreatora zmodyfikować zapytanie. Można je otworzyć w Microsoft Query i dokonać zaawansowanej edycji. Jeśli zapytanie jest zaawansowane, pojawia się komunikat o braku możliwości zmiany za pomocą kreatora i jest otwierane w Microsoft

Query.

Utworzone połączenie do bazy można wykorzystać przy tworzeniu kolejnych zapytań.



2.17. Zapytania do danych geograficznych (OPCJONALNE)

Wyświetl dane z tabeli **dbo.Province** bazy danych **Spatial**.

W oknie zapytania zostaną wyświetlone zawarte w tabeli dane, w tym binarne dane z kolumny o nazwie **geom**, które zawierają dane geograficzne (podobne do odczytywanych przez GPS).

SQLQuery5.sql - JM5QL2012\...

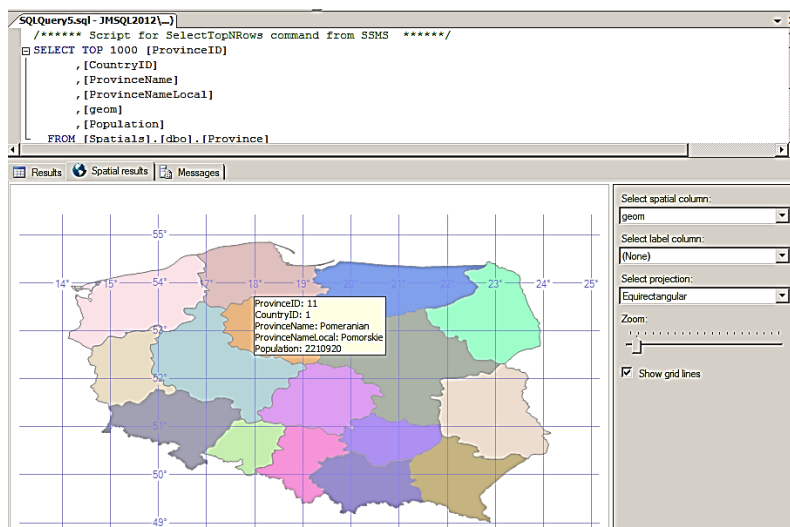
```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP 1000 [ProvinceID]
, [CountryID]
, [ProvinceName]
, [ProvinceNameLocal]
, [geom]
, [Population]
FROM [Spatial]. [dbo]. [Province]

```

	ProvinceID	CountryID	ProvinceName	ProvinceNameLocal	geom	Population
1	1	1	Greater Poland	Wielkopolskie	0xE610000001041403000048C5E3A25ACC4A4000D57954FCE...	3386882
2	2	1	Kuyavian-Pomeranian	Kujawsko-Pomorskie	0xE610000001040B020000A877F17EDCD84A4050E50AEF724...	2066136
3	3	1	Lesser Poland	Małopolskie	0xE610000001045E040000B87A87DBA1374940A03BDEE4B7...	3279036
4	4	1	Lodz	Łódzkie	0xE610000001041A02000008B5C35F93274A4070200890A157...	2555898
5	5	1	Lower Silesian	Dolnośląskie	0xE610000001048E070000B80FB22C98D4494090AE0B3F382...	2878410
6	6	1	Lublin	Lubelskie	0xE61000000104C90600004874081C09264A40D080B3942C33...	2166213
7	7	1	Lubusz	Lubuskie	0xE610000001043803000070F59CF4BE8D4A40C08CF161F6D...	1008481
8	8	1	Masovian	Mazowieckie	0xE610000001046303000058F564FED1AF4A4010FDF7E0B58...	5188488
9	9	1	Opole	Opolskie	0xE6100000010486030000881A3048FA9049404089EC832CCB...	1037088
10	10	1	Podlachian	Podlaskie	0xE610000001048D030000D083CBB73E364B4050B75ED383...	1192660
11	11	1	Pomeranian	Pomorskie	0xE61000000107C72800005AB37ED7BC2E4B40A7FE2F0874C...	2210920
12	12	1	Silesian	Śląskie	0xE6100000010420040000F89579ABAEB14940F001D71533FA...	4654115
13	13	1	Subcarpathian	Podkarpackie	0xE610000001041206000010BD1AA034624940C0D0402C9B0...	2097338
14	14	1	Swietokrzyskie	Świętokrzyskie	0xE6100000010481010000103B53E8BC924940604F58E201B5...	1275550
15	15	1	Warmian-Masurian	Warmińsko-Mazurs...	0xE61000000104FB070000C8718BF9B9374B40C08D78B29BC...	1426155
16	16	1	West Pomeranian	Zachodniopomorskie	0xE61000000107D73100007625668767D14A40A3A4986B132...	1692271

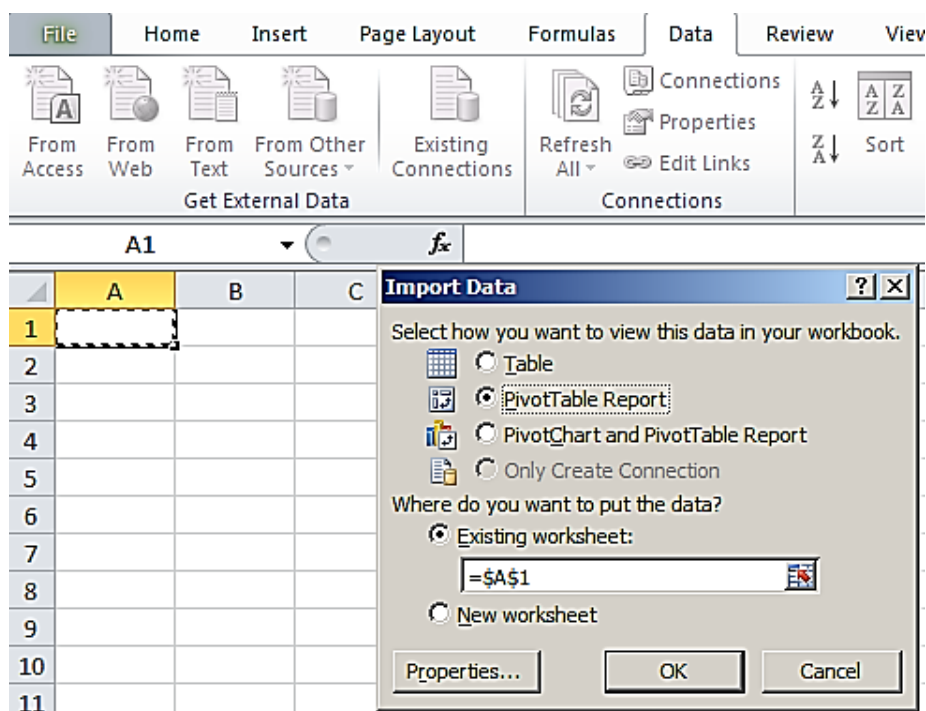
Wybierz zakładkę **Spatial results**, żeby wyświetlić mapę województw Polski.



2.18. Wykonywanie raportów (OPCJONALNE)

Wykonaj raport tabeli przestawnej.

- W nowym skoroszybie Microsoft Excel Wybierz zakładkę (wstążkę) **Data**,
- Wybierz menu **From Other Sources**, a następnie polecenie **From SQL Server**,
- Ustaw połączenie do bazy **Northwind**, do tabeli **V_Sprzedane**,
- Importuj dane do raportu tabeli przestawnej (opcja **PivotTable Report**) ,



Kolumny **Quantity** , **UnitPrice** umieść jako pola wartości, zaś kolumnę **ProductName** w polu wierszy.

3. Praktyczny przykład zastosowania .

Waszym zadaniem jest rozwiązanie przykładowego problemu (CS) pracując w grupie, stosując burzę mózgow oraz pytań i odpowiedzi wg. schematu:

- d) Zapoznajecie się z prezentowanym problemem,
- e) Analizujecie problem,
- f) Wykonujecie zadane polecenia,
- g) Prezentujecie wypracowany materiał.

2.1. Case Study Projekt bazy danych dla dziennika elektronicznego

Zadanie do wykonania

- a) Przeanalizujcie w grupie, jakie informacje są potrzebne w dzienniku elektronicznym
- b) Pogrupujcie te informacje w encje (informacje jednego rodzaju np. przedmiot, nauczyciel, uczeń), jakie atrybuty powinna mieć każda z encji (np. nazwisko, imię)
- c) Zdefiniujcie zależności pomiędzy encjami
- d) Przeanalizujcie przykładową bazę Dziennik
- e) Jakie usprawnienia warto wprowadzić do bazy Dziennik?

4. Załączniki

- 1. Pliki przykładowych baz: Northwind (przystosowana do Ms SQL 2012) i Spatial.
- 2. Skrypt tworzący bazę Dziennik – Dziennik.sql.
- 3. Pliki w podfolderze .\PlikiDoBazyDziennik\
 - a. Klasy.xlsx
 - b. Nauczyciele.xlsx
 - c. Nazwiska_Imiona.xlsx

Załączniki te znajdują się w lokalizacjach:

- 1) na fizycznym serwerze w folderze c:\Nauczyciele\SQL
- 2) Spakowane w obraz CD w pliku c:\nauczyciele\SQL.iso
- 3) W wirtualnej maszynie JMSQL2012 w folderze c:\Nauczyciele

5. Literatura i wykaz linków

- 1. Linki:
 - a) Books Online for SQL Server 2012:

- <http://msdn.microsoft.com/en-us/library/ms130214.aspx>
- b) Instalacja SQL Server 2012 Express with SP1
<http://msdn.microsoft.com/en-us/evalcenter/hh230763.aspx>
- c) Instalacja Microsoft SQL Server 2008 R2 RTM - Express with Advanced Services
<http://www.microsoft.com/en-us/download/details.aspx?id=25174>
- d) Transact-SQL Reference (Database Engine)
<http://msdn.microsoft.com/en-us/library/bb510741.aspx>
- e) Data Types (Transact-SQL)
<http://msdn.microsoft.com/en-us/library/ms187752.aspx>
- f) Types of Spatial Data
[http://msdn.microsoft.com/en-us/library/bb964711\(v=sql.105\).aspx](http://msdn.microsoft.com/en-us/library/bb964711(v=sql.105).aspx)
- g) SQL Server Spatial Tools
<http://sqlspatialtools.codeplex.com/>
<http://www.sharpgis.net/page/sql-server-2008-spatial-tools.aspx>
- h) Shapefile maps
<http://www.statsilk.com/maps/download-free-shapefile-maps>
- i) Reporting Services (SSRS)
<http://msdn.microsoft.com/en-us/library/ms159106.aspx>