

Adam Pelikant

MS SQL Server

ZAAWANSOWANE METODY PROGRAMOWANIA
WYDANIE II

Helion 

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz wydawca dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz wydawca nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redaktor prowadzący: Grzegorz Krzystek
Projekt okładki: Studio Gravite

Helion S.A.
ul. Kościuszki 1c, 44-100 Gliwice
tel. 32 231 22 19, 32 230 98 63
e-mail: helion@helion.pl
WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
<http://helion.pl/user/opinie/sqlsz2>
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Materiały dodatkowe do książki znajdziesz pod adresem:
<https://ftp.helion.pl/przyklady/sqlsz2.zip>

ISBN: 978-83-283-7412-6

Copyright © Helion S.A. 2022

Printed in Poland.

- [Kup książkę](#)
- [Poleć książkę](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

Spis treści

Od autora (do wydania I)	5
Od autora (do wydania II)	8
Rozdział 1. Podstawowe informacje o serwerze	11
Rozdział 2. Instalacja i konfiguracja środowiska	15
Rozdział 3. Język zapytań SQL w MS SQL Server	38
3.1. Zapytania wybierające	38
3.2. Zapytania modyfikujące dane	102
3.3. Tworzenie i modyfikacja tabel i perspektyw	112
3.4. Modyfikowanie tabel	166
3.5. Perspektywy (widoki)	176
3.6. Tworzenie typu użytkownika	199
3.7. Tworzenie indeksów	203
3.8. Inne narzędzia klienckie MS SQL Server	220
Rozdział 4. Problemy rozwiązywane z wykorzystaniem SQL	224
Rozdział 5. Rozszerzenia proceduralne Transact-SQL	263
5.1. Podstawowe instrukcje	263
5.2. Procedury składowane	270
5.3. Funkcje	279
5.4. Synonimy i błędy użytkownika	284
5.5. Procedury wyzwalane	291
5.6. Kursory	329
5.7. Zmienna tabelaryczna i typ tabelaryczny	349

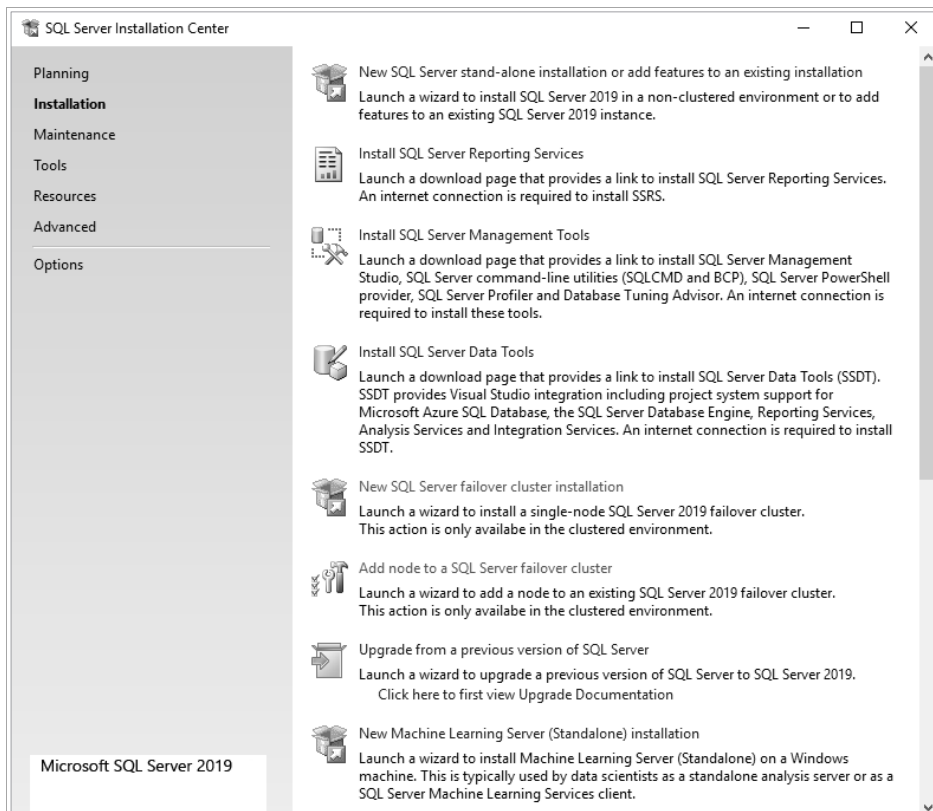
Rozdział 6. Przetwarzanie transakcyjne	354
6.1. Transakcje. Podstawy teoretyczne	354
6.2. Transakcje. Przykłady realizacji	358
6.3. Obsługa wyjątków	367
Rozdział 7. Typy złożone	375
7.1. Typ tabelaryczny	375
7.2. Typ hierarchiczny	380
7.3. Tabele grafowe	389
7.4. Typy geometry i geography	396
7.5. Typy użytkownika CLR	413
7.6. Elementy proceduralne CLR	433
Rozdział 8. Problemy rozwiązywane za pomocą rozszerzenia proceduralnego i obiektowego	461
8.1. Klasyfikacja	461
8.2. Funkcje agregujące definiowane przez użytkownika	474
8.3. Analiza sieci powiązań	489
Zakończenie	502
Literatura	504

Rozdział 2.

Instalacja i konfiguracja środowiska

Proces instalacji MS SQL Server jest dość intuicyjny i nie wymaga bardzo szczegółowego omówienia. Dla porządku jednak zostaną przedstawione najistotniejsze kroki instalacji, ze szczególnym podkreśleniem tych, przy których mniej doświadczony użytkownik może mieć pewne wątpliwości. Po uruchomieniu instalatora pojawia się okno o nazwie *SQL Server Installation Center*, które zawiera zakładki:

- *Planning*, zawierającą elementy związane z wymaganiami oraz dostępną dokumentacją:
 - ✓ *Hardware and Software Requirements*,
 - ✓ *Security Documentation*,
 - ✓ *Online Realise Notes*,
 - ✓ *System Configuration Checker*,
 - ✓ *Download Data Migration Assistant (DMA)*,
 - ✓ *Online Installation Help*,
 - ✓ *How to Get Started with SQL Server 2019 Failover Clustering*,
 - ✓ *Upgrade Documentation*,
 - ✓ *Install SQL Server Migration Assistant (SSMA)*,
 - ✓ *How to apply SQL Server updates*;
- *Installation* (rysunek 2.1), która jest podstawową zakładką w procesie instalacji, zawierającą elementy:
 - ✓ *New SQL Server stand-alone installation or add features to an existing installation* — instalacja nowej instancji serwera albo dodanie funkcjonalności do już zainstalowanej,



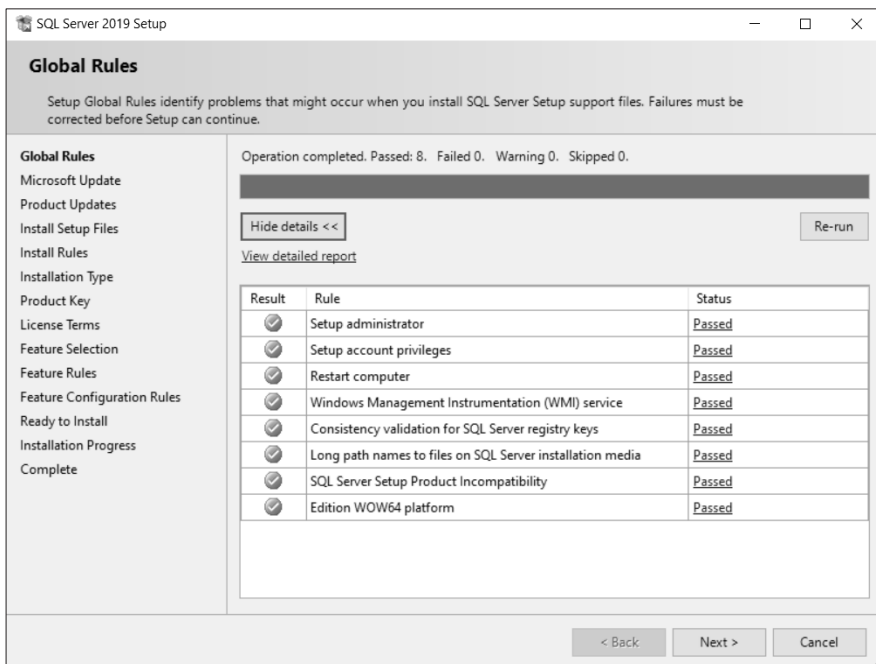
RYСУNEK 2.1. Instalacja SQL Server 2019 — podstawowa zakładka

- ✓ *Install SQL Server Reporting Services* — instalacja środowiska do tworzenia i przetwarzania raportów wraz z serwerem http,
- ✓ *Install SQL Server Management Tools* — instalacja standardowej, wizualnej końcówki klienckiej,
- ✓ *Install SQL Server Data Tools* — instalacja narzędzi do integracji danych oraz przetwarzania analitycznego (hurtowni danych),
- ✓ *New SQL Server failover cluster* — nowa instancja węzła w przypadku konfiguracji w postaci klastra (przetwarzanie w gridzie — sieci),
- ✓ *Add node to a SQL Server failover cluster* — dodanie węzła do istniejącej instancji klastra,
- ✓ *Upgrade from a previous version of SQL Server* — aktualizacja starszej wersji instancji serwera do wersji aktualnej,

- ✓ *New Machine Learning Server (Standalone) installation* — dodanie narzędzi do uczenia maszynowego, umożliwienie stosowania języków skryptowych R oraz Python;
- *Maintenance* (zarządzanie), zawierająca elementy:
 - ✓ *Edition Upgrade* — aktualizacja istniejącej instancji serwera do bardziej ogólnej (wyższej) edycji,
 - ✓ *Repair* — naprawienie zainstalowanej instancji serwera,
 - ✓ *Remove node from a SQL Server failover cluster* — usunięcie węzła z instancji klastra,
 - ✓ *Launch Windows Update to search for product updates* — wyszukiwanie uaktualnień dla zainstalowanej instancji bazy danych;
- *Tools*, zawierająca elementy:
 - ✓ *System Configuration Checker* — sprawdzenie zgodności ze stanem faktycznym wymagań systemowych dla instalacji serwera,
 - ✓ *Installed SQL Server features discovery report* — wyświetlenie informacji o dostępnych funkcjonalnościach instancji serwera,
 - ✓ *Microsoft Assessment and Planning (MAP) Toolkit for SQL Server* — wspomaganie migracji między serwerami baz danych różnych producentów;
- *Resources*, pozwalającą na dostęp do informacji technicznej i zawierającą elementy:
 - ✓ *SQL Server 2019 Books Online*,
 - ✓ *SQL Server TechCenter*,
 - ✓ *SQL Server Developer Center*,
 - ✓ *SQL Server Product Evaluation Web site*,
 - ✓ *License agreement*,
 - ✓ *Register your copy of SQL Server 2019 Express*,
 - ✓ *SQL Server Privacy Statement*,
 - ✓ *Community*,
 - ✓ *SQL samples Web site*;
- *Advanced*, stanowiącą zestaw zaawansowanych narzędzi konfiguracyjnych i zawierającą elementy:
 - ✓ *Install base on configuration file* — instalacja serwera na podstawie wcześniej utworzonego pliku konfiguracyjnego,

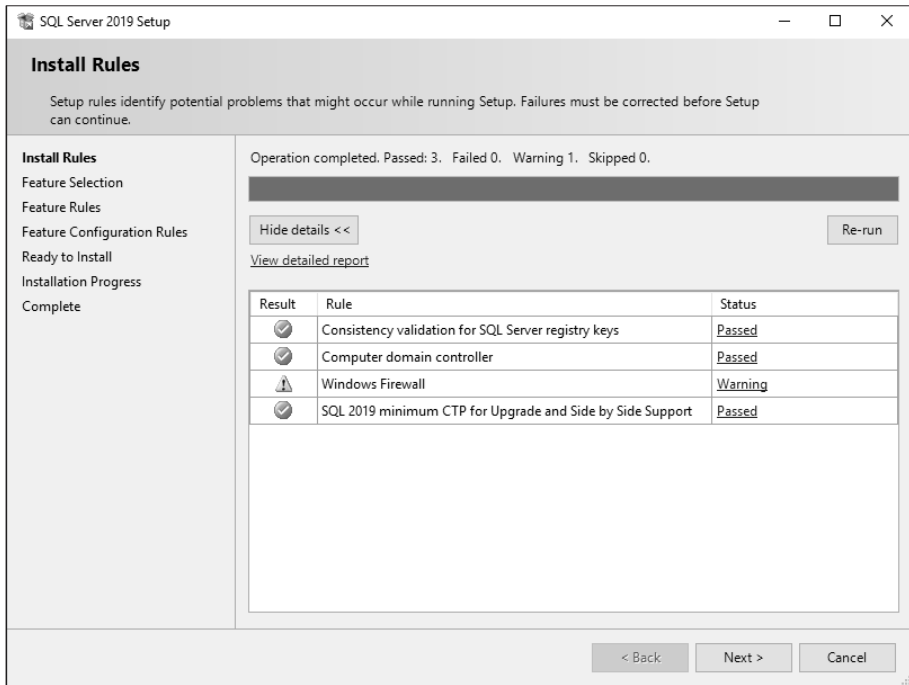
- ✓ *Advanced cluster preparation* — zaawansowane opcje instalacji serwera do postaci klastra,
 - ✓ *Advanced cluster completion* — dokończenie instalacji serwera do postaci klastra,
 - ✓ *Image preparation of a stand-alone instance of SQL Server* — przygotowanie obrazu na podstawie instancji zainstalowanego serwera,
 - ✓ *Image completion of a prepared stand-alone instance of SQL Server* — dokończenie tworzenia obrazu na podstawie instancji zainstalowanego serwera;
- *Options*, pozwalająca na wskazanie napędu (folderu) zawierającego źródła do instalacji (domyślnie ustawiany na miejsce, z którego uruchomiono aplikację instalatora).

Po wybraniu podstawowej, pojedynczej, nowej instancji SQL Server pojawia się okno, w którym dokonujemy wyboru instalowanej edycji (w przypadku licencji MSDN — *Developer*), a dla licencji profesjonalnych możemy podać klucz produktu. W kolejnym kroku zostaniemy zapoznani z warunkami licencji, które musimy zaakceptować, aby przejść do kolejnego etapu. Następnie sprawdzana jest zgodności formalnych wymagań systemowych ze stanem rzeczywistym — rysunek 2.2.



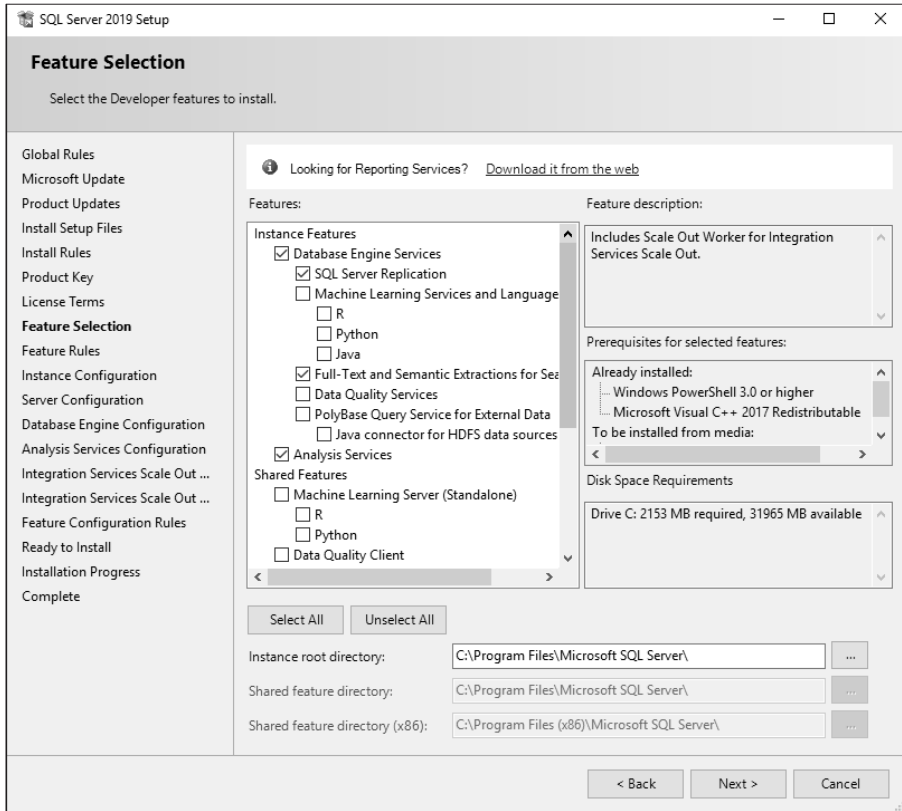
RYСУNEK 2.2. Zakończenie procesu weryfikacji wymagań programu

Po pierwszym procesie sprawdzenia następują kolejne weryfikacje, określane jako: *Microsoft Update*, *Product Updates*, a jeżeli aktualizacje zostaną znalezione, następuje ich pobieranie. Kolejna weryfikacja dotyczy wykrywania potencjalnych problemów podczas instalacji — rysunek 2.3. Gdy weryfikacja zakończy się pozytywnie, w kolejnym oknie dokonujemy wyboru zakresu prowadzonej instalacji — rysunek 2.3. Jako rezultat tego etapu często pojawia się ostrzeżenie dotyczące zapory sieciowej systemu. W przypadku instalacji dla potrzeb edukacyjnych i testowych nie stanowi ono problemu, jednakże dla rozwiązań komercyjnych konieczna jest bezwarunkowa zmiana ustawień zapory. Pozostałe wymagania powinny zostać pozytywnie zweryfikowane.



RYSUNEK 2.3. Sprawdzenie wymagań dotyczących środowiska

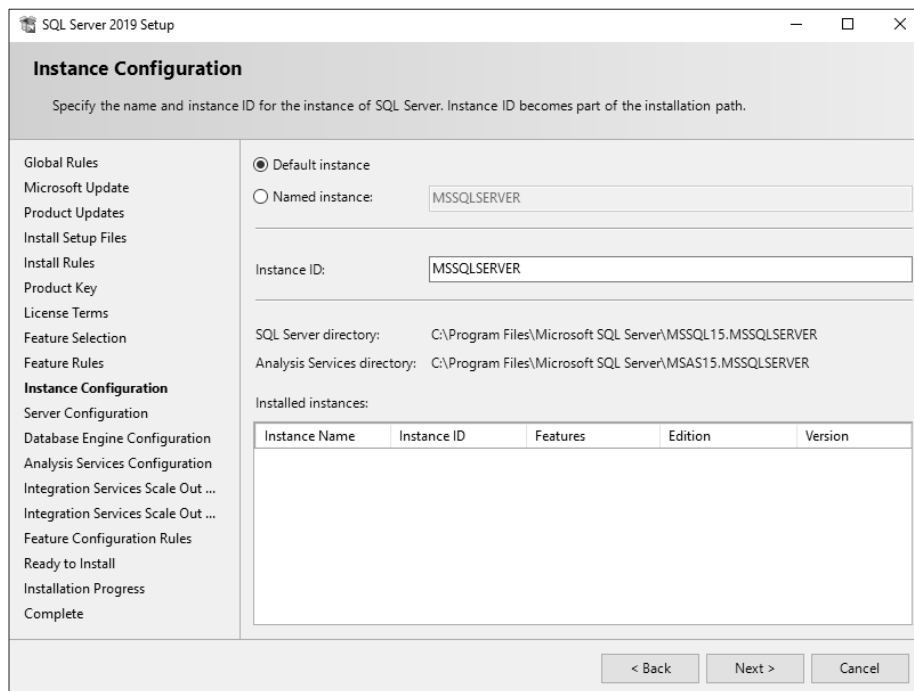
W następnym kroku (rysunek 2.4) ustalany jest szczegółowy zakres instalacji. Wybieramy w nim te komponenty, które zostaną zainstalowane. Oczywiście podstawą jest silnik bazy danych (*Database Engine*), ale wskazane jest zainstalowanie również silnika analitycznego. Warto rozważyć dodanie narzędzi raportujących, jednak jak to pokazuje informacja w górnej części okna dialogowego, konieczne jest ściągnięcie ich z sieci. W przykładzie zdecydowano się na dodanie narzędzia do replikacji oraz wyszukiwania pełnotekstowego. Wybór zakresu instalowanych komponentów zależy od Czytelnika. Zwrócić należy uwagę na zasoby sprzętowe, jakimi dysponujemy, a w przypadku języków Python i R należy mieć już zainstalowane silniki tych środowisk. Należy pamiętać, że w trakcie korzystania z MS SQL Server można za pomocą tego samego instalatora dodać do działającej instancji kolejne składniki.



RYСУNEK 2.4. Wybór komponentów do zainstalowania

Gdy dokonamy wyboru, w kolejnym kroku następuje weryfikacja wymagań wynikających z wybranych komponentów, która nie została przedstawiona graficznie. Jeśli zasoby nie będą wystarczające, wymagane jest wybranie innej lokalizacji albo zredukowanie liczby instalowanych elementów.

Kolejny etap to ustalenie nazwy instalowanej instancji (rysunek 2.5). Kiedy instalujemy pierwszą instancję, wskazane jest pozostawienie opcji *Default instance* — co ustala jej nazwę na *MSSQLSERVER*. Ponieważ na jednym komputerze może być zainstalowanych wiele instancji, w przypadku instalowania kolejnej konieczne jest nadanie innej, niedomyślnej nazwy. W tym samym oknie dialogowym możemy ustawić inną niż domyślna ścieżkę do katalogu, w którym będzie odbywała się instalacja, oraz uzyskać informację o wcześniej zainstalowanych instancjach serwera. Niekiedy użytkownik może być zaskoczony tym, że pomimo iż nie instalował świadomie wcześniej żadnych instancji serwera, pojawia się informacja o już zainstalowanych komponentach. Dzieje się tak, gdy zostało zainstalowane środowisko .NET, dla którego domyślnym składnikiem jest SQL Server w wersji *Express*.

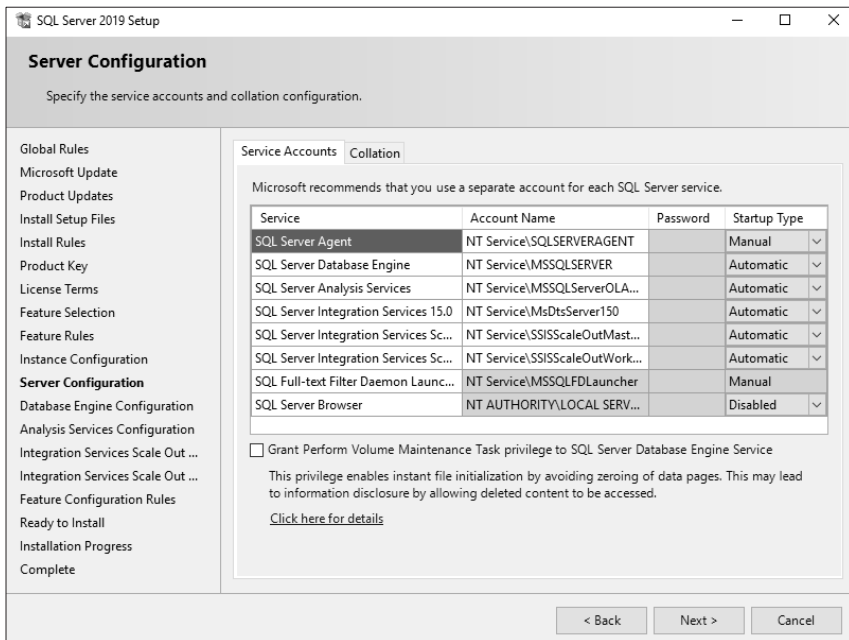


RYСУNEK 2.5. Definicja instancji serwera

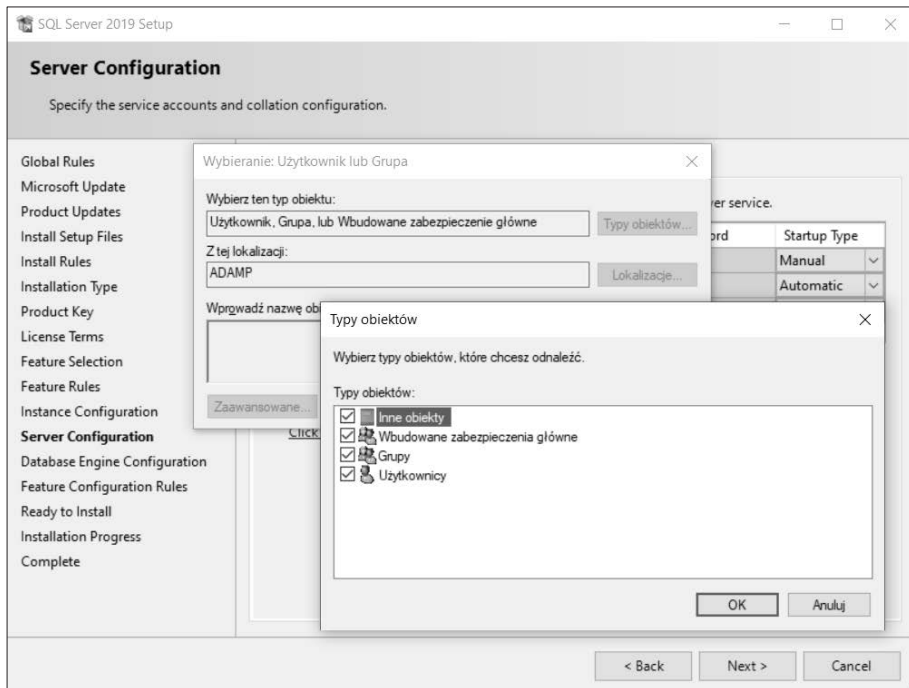
Po tym kroku następuje sprawdzenie wymagań związanych z zasobami dyskowymi. Tak samo jak poprzednio, jeśli zasoby okażą się niewystarczające, należy albo zmienić lokalizację (dysk), albo cofając się do właściwego okna dialogowego, zmniejszyć liczbę instalowanych komponentów.

Kolejny etap to wybór konta, na rzecz którego będą uruchamiane poszczególne serwisy serwera, oraz trybu ich uruchomienia (automatyczny, ręczny) — rysunek 2.6. Ten etap konfiguracji nastrocza sporo problemów, ponieważ wydaje się, że najlepszym wyborem będzie ustawienie konta lokalnego administratora. W takim przypadku usługi nie będą włączały się automatycznie po uruchomieniu systemu. Dlatego najrozsądniejsze wydaje się przypisanie praw do uruchomienia serwisów jednej spośród usług, która jest automatycznie uruchamiana podczas startu systemu operacyjnego i w konsekwencji automatycznie uruchomi serwisy bazy danych.

W starszych wersjach dla większości serwisów do wyboru mieliśmy: usługę sieciową, usługę lokalną i system. Obecnie każdy serwis ma dedykowaną usługę i pomimo że można skorzystać z wyboru innej usługi za pomocą pozycji <<Browse>> (rysunek 2.7), to zalecam pozostawienie stanu domyślnego, który zapewnia poprawne funkcjonowanie środowiska.



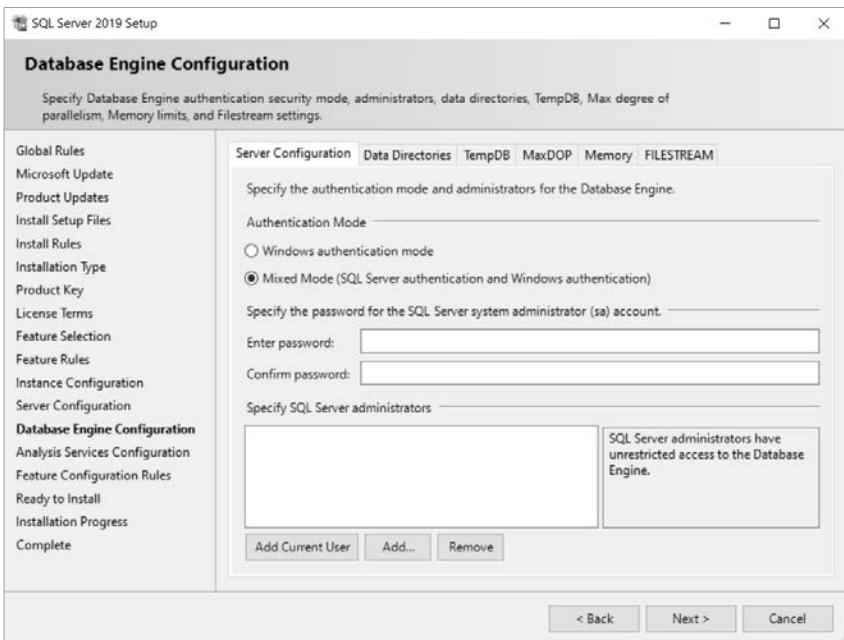
RYSUNEK 2.6. Wybór sposobu uruchamiania usług



RYSUNEK 2.7. Wybór niedomyślnych opcji uruchamiania usług serwera

W starszych wersjach możliwe było jednoczesne przypisanie jednego serwisu do wszystkich usług za pomocą przycisku *Use the same account for all SQL Server services*, z czego w wersji 2012 zrezygnowano, co podkreśla zasadność stosowania ustawień domyślnych. Po zatwierdzeniu wyboru pozostaje tylko ustanowienie, które z usług będą mimo wszystko uruchamiane ręcznie. W przypadku dużej pamięci RAM można pozostawić ustawienia domyślne. Natomiast przy niewielkich zasobach proponuję pozostawienie tylko automatycznego uruchamiania serwisu silnika bazy danych. Pozostałe serwisy będą wtedy uruchamiane ręcznie za pomocą narzędzi zarządzania komputerem — pozycja *Usługi*.

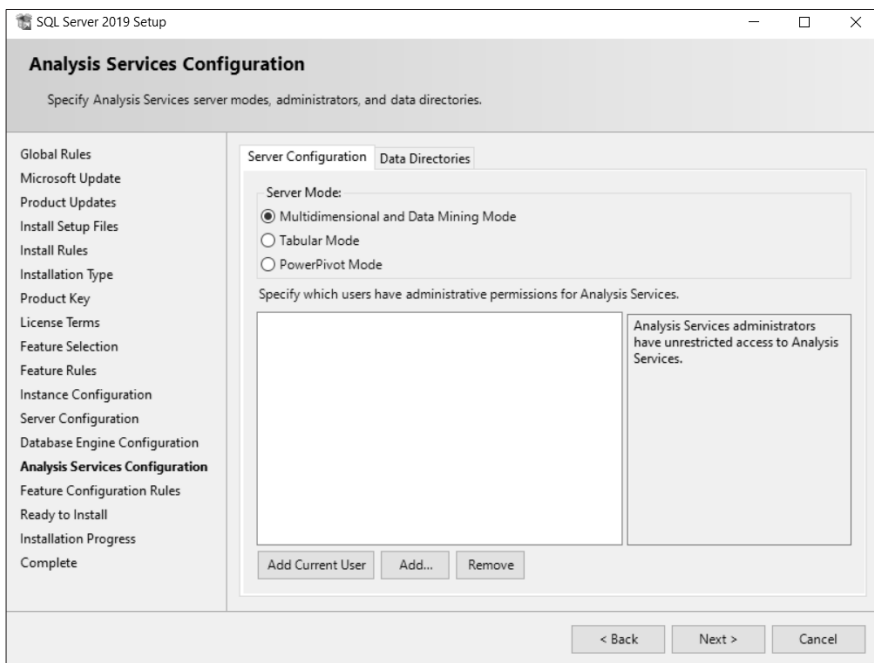
Kolejnym etapem jest ustalenie dostępnych trybów uwierzytelnienia (rysunek 2.8). Stanem domyślnym jest autoryzacja za pomocą systemu operacyjnego — *Windows authentication mode*. Dostępny jest wówczas tylko ten tryb uwierzytelnienia. Drugi stan to *Mixed Mode*, który pozwala na ustawienie dwóch trybów uwierzytelnienia: opartego na systemie operacyjnym oraz niezależnego od uwierzytelnienia w Windows trybu autoryzacji. Proponuję ustanowienie tego drugiego sposobu. W tym przypadku należy ustalić hasło dla tworzonego w tym trybie superadministratora o nazwie *sa*. W starszych wersjach MS SQL Server każdy administrator Windows stawał się automatycznie administratorem serwera bazy danych. Obecnie w kontrolce *Specify SQL Server administrators* należy podać tych użytkowników lub ich grupy, którzy lub które otrzymają takie uprawnienia. Możemy użyć przycisku *Add Current User*, który nadaje takie prawa bieżącemu (zalogowanemu w czasie procesu instalacji) użytkownikowi Windows, albo *Add...*, który pozwala na dokonanie wyboru innego użytkownika lub grupy.



RYСУNEK 2.8. Wybór trybów autoryzacji do SQL Server

Proponuję w tym miejscu dodać przynajmniej grupę lokalnych administratorów lub administratorów domenowych. Zestaw użytkowników posiadających uprawnienia do logowania się w SQL Server bez podania hasła i tylko na podstawie poprawnego zalogowania do systemu może być dowolnie długi.

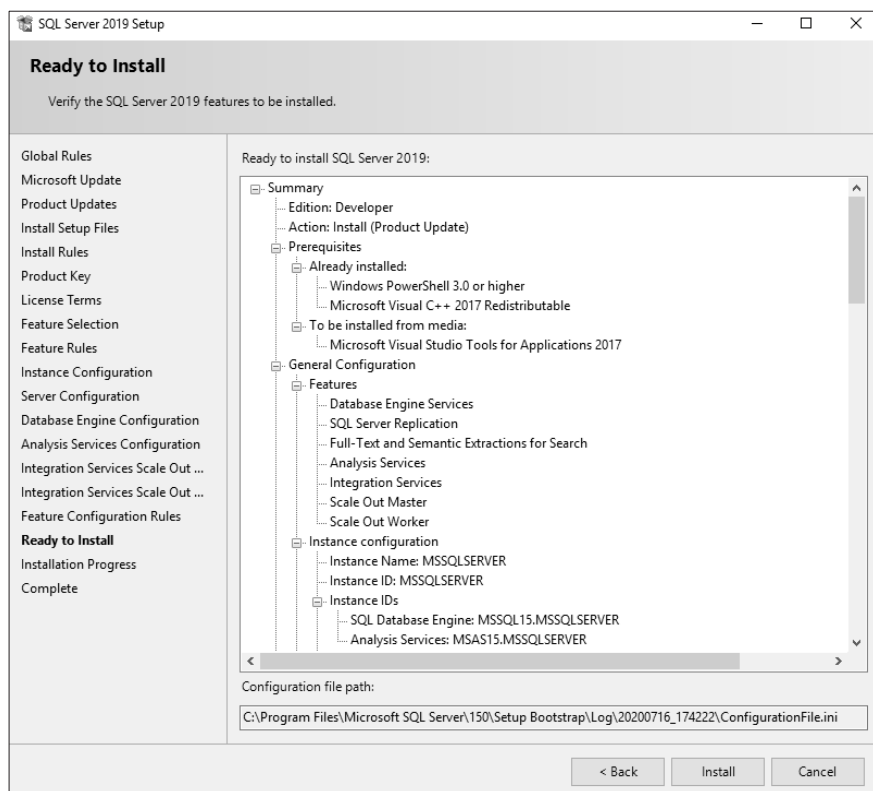
Podobnego wyboru musimy dokonać dla serwisu analitycznego (rysunek 2.9) — *Analysis Services Configuration*. Podobnie jak w przypadku silnika bazy danych, proponuję dodać bieżącego użytkownika i grupę lokalnych administratorów systemu. Domyślnym stanem jest instalacja modelu tabelarycznego. Z punktu widzenia zagadnień poruszanych w tej książce można jej nie zmieniać, ja jednak preferuję model wielowymiarowy.



RYСУNEK 2.9. Wybór autoryzacji do *Analysis Services*

W kolejnym etapie następuje podsumowanie wszystkich wybranych ustawień, które są widoczne w postaci strony WWW oraz w ostatnim oknie dialogowym instalatora (rysunek 2.10), a po zatwierdzeniu następuje proces instalacji. Po pomyślnym jego zakończeniu powinniśmy w menu *Start* mieć dostępne podstawowe komponenty i narzędzia SQL Server.

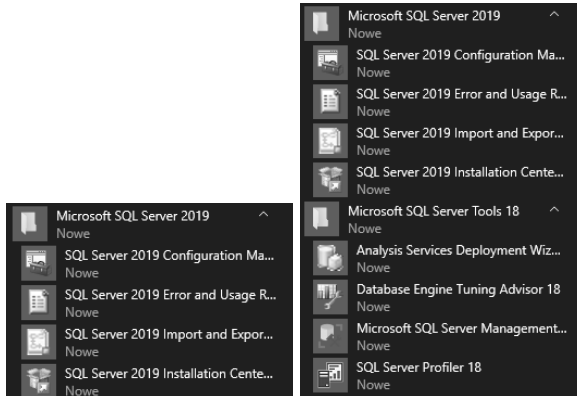
Na rysunku 2.11 przedstawiona została rozwinięta grupa w środowisku Windows 10. Niestety nie ma wśród nich narzędzia do zarządzania oraz uruchamiania zapytań i skryptów *SQL Server Management Studio*, które musi być zainstalowane oddzielnie. W tym celu musimy ponownie uruchomić aplikację instalacyjną, lecz tym razem należy wybrać pozycję *Install*



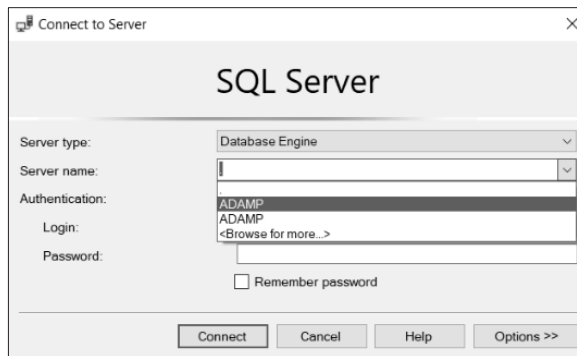
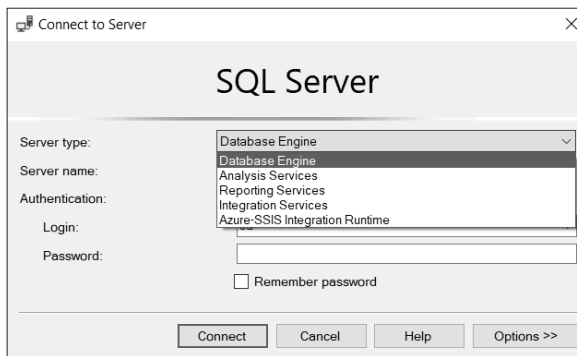
RYСУNEK 2.10. Podsumowanie informacji o instalowanych komponentach MS SQL Server

SQL Server Management Tools (rysunek 2.1). Spowoduje to przeniesienie nas do strony WWW, z której należy pobrać plik instalatora. W przypadku opisywanego procesu w wersji 18.5.1 (SSMS-Setup-ENU.exe) prawdopodobnie w momencie instalacji Czytelnik będzie miał do dyspozycji nowszą realizację tego oprogramowania, ponieważ prace nad serwerem i końcówką kliencką realizowane są w dużej mierze niezależnie. Proces instalacji *Management Studio* jest prosty i nie zostanie precyzyjniej opisany. Należy jednak zwrócić uwagę na fakt, że w menu *Start* pojawi się dodatkowa grupa Microsoft SQL Server Tools 18, w której znajdują się dodatkowe komponenty — między innymi *Management Studio* (rysunek 2.11). Na podobnych zasadach możliwe jest zainstalowanie *Reporting Services*.

Jeśli uruchomimy *SQL Server Management Studio*, to jako pierwsze pojawi się okno logowania (rysunek 2.12), które zawiera kilka kontrolek. Pierwsza z nich pozwala na określenie silnika, z którym będziemy się łączyli. Dostępne jest połączenie z: serwerem bazy danych (*Database Engine*), hurtownią danych (*Analysis Services*), systemem raportującym (*Reporting Services*), systemem integracji danych (*Integration Services*) oraz migracją do chmury (*Azure SSIS Integration Runtime*). Jeśli wybierzemy serwer danych, to konieczne będzie określenie nazwy tego, z którym będziemy się łączyli. Możliwe jest ręczne wpisanie nazwy albo wybranie



RYSUNEK 2.11. Komponenty MS SQL Server przed i po zainstalowaniu Management Studio

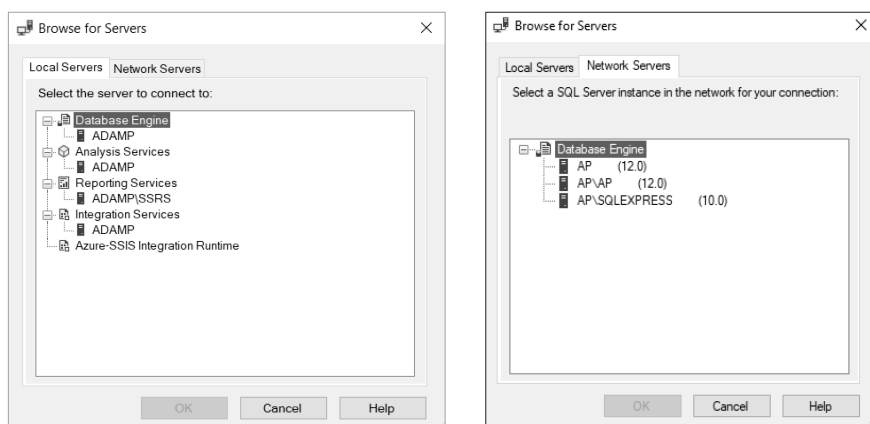


RYSUNEK 2.12. Wybór rodzaju silnika oraz instancji serwera

serwera z listy. Domyślną nazwą serwera jest nazwa hosta (komputera), na którym jest on zainstalowany — w pokazywanym przypadku jest to *ADAMP*. Jeśli jednak chcemy połączyć się z serwerem lokalnym (zainstalowanym na komputerze, z którego następuje logowanie), możemy użyć nazwy logicznej . (kropka). Różnica polega na tym, że w przypadku podania

nazwy hosta serwer rozgłasza żądanie obsługi w sieci, a następnie „sam sobie odpowiada”, że jest tym hostem, z którym chcieliśmy się połączyć. Jeśli jawnie podamy, że jest to ten sam komputer (kropka), proces rozgłaszania nie jest potrzebny. Druga z definicji połączenia pozwala na znaczne ograniczenie ruchu w sieci.

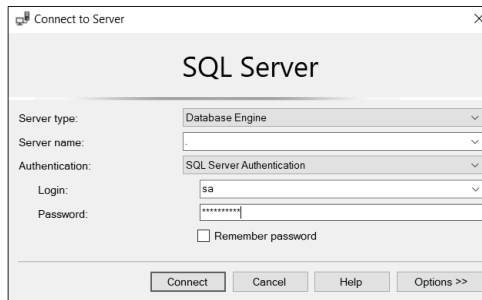
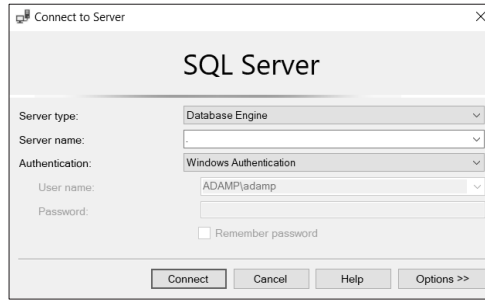
Na rysunku 2.12 poza kropką oraz nazwą hosta ostatnią pozycją na liście jest <Browse for more...>; konsekwencją jej wybrania jest pojawienie się okna dialogowego — rysunek 2.13.



RYСУNEK 2.13. Wykrywanie instancji serwerów lokalnych i zainstalowanych w domenie (drugi przypadek dotyczy innej lokalizacji serwera)

Okno to zawiera dwie zakładki. Pierwsza (domyślna) pokazuje wszystkie zainstalowane lokalnie silniki, zarówno bazy danych (pełnej oraz EXPRESS, jeśli zainstalowano), jak i wszystkich innych dostępnych narzędzi. Przejście do drugiej zakładki powoduje proces wykrywania serwerów baz danych zainstalowanych w domenie (grupie roboczej), do której należy nasz komputer. Aby serwer został wykryty, musi być uruchomiony komputer oraz serwis silnika bazy danych. Widoczne obok nazw liczby określają wersję serwera: 15.0 — wersja 2019, 14.0 — wersja 2017, 13.0 — wersja 2016, 12.0 — wersja 2014, 11.0 — wersja 2012, 10.0 — wersja 2008, w tym 2008 R2, 9.0 — wersja 2005; pojawienie się liczby 8.0 oznaczałoby wersję 2000. Możliwe jest dokonanie wyboru dowolnego serwera z obu zakładek okna dialogowego i zalogowanie się do niego, pod warunkiem że umiemy się uwierzytelnić. Jak widać, korzystając z lokalnego *SQL Server Management Studio*, możemy zarządzać dowolnym serwerem w domenie.

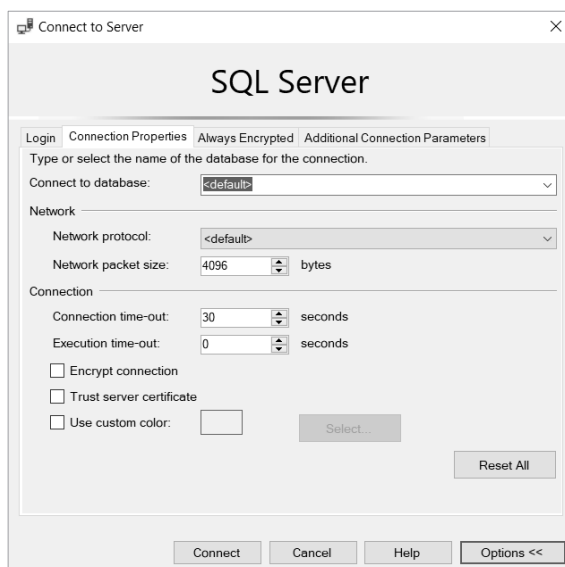
Jeżeli dokonamy wyboru serwera (w pokazanym przykładzie wybrano serwer lokalny), pozostaje ustalenie trybu autoryzacji — rysunek 2.14. Do uwierzytelnienia systemowego *Windows Authentication* nie jest potrzebne podawanie żadnych dodatkowych danych (zarówno nazwa użytkownika, jak i hasło zostały zweryfikowane podczas logowania do Windows), natomiast dla *SQL Server Authentication* konieczne jest podanie zarówno nazwy użytkownika, jak i hasła. W przykładzie użyte zostały konto domyślnego superadministratora *sa* oraz hasło zdefiniowane podczas instalacji.



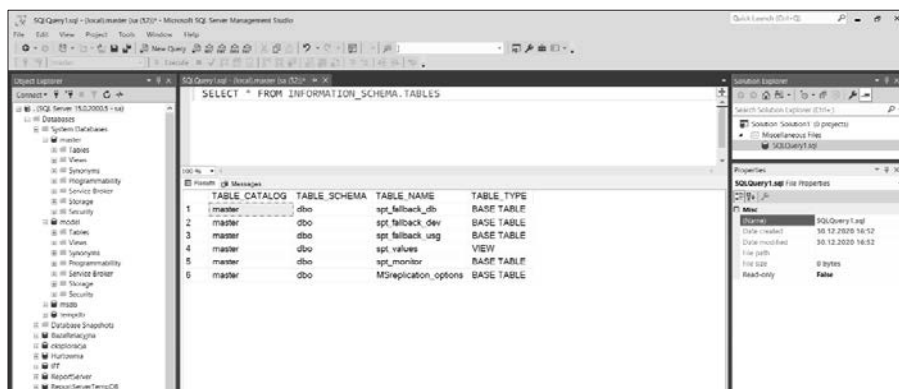
RYСУNEK 2.14. Rodzaje autoryzacji do serwera

Poza danymi do logowania możliwe jest określenie dodatkowych właściwości (rysunek 2.15), takich jak: czas na połączenie (*Connection time-out*), ograniczenie czasu wykonania poleceń SQL (*Execution time-out*), zastosowanie szyfrowania połączenia (*Encrypt connection*). W postaci list rozwijanych dostępne są kolejne dwa atrybuty: *Connect to database* z dopuszczalnymi opcjami *<default>* i *<Browse server...>* oraz *Network protocol* z dopuszczalnymi opcjami *<default>*, *Shared Memory*, *TCP/IP*, *Named Pipes*. Dla każdego protokołu możliwe jest ustalenie wielkości pakietu *Network packet size*.

Po pomyślnym procesie logowania widzimy wewnątrz *Microsoft SQL Server Management Studio* — rysunek 2.16. Lewy panel okna zawiera przedstawioną w postaci drzewa hierarchicznego strukturę instancji serwera (na rysunku została ona częściowo rozwinięta). Po prawej stronie w górnej części widoczne jest okno służące do tworzenia i uruchamiania zapytań i skryptów SQL. Pośrodku znajduje się dwuzakładkowa kontrolka służąca do wyświetlania rezultatów zapytań. Na rysunku 2.16 widoczna jest zakładka *Results*, zawierająca zestaw rekordów zwróconych przez zapytanie, z tyłu widać zakładkę *Messages*, gdzie pojawiają się komunikaty z bazy.



RYSUNEK 2.15. Zaawansowane ustawienia właściwości połączenia



RYSUNEK 2.16. Widok struktury serwera oraz panelu przetwarzania zapytań w Microsoft SQL Server Management Studio

W strukturze hierarchicznej widoczne są cztery występujące zawsze bazy systemowe:

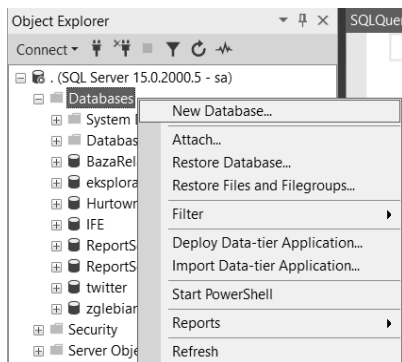
- *master* — główna baza systemowa, zawierająca wszystkie obiekty systemowe (tabele, perspektywy, procedury etc.); na bazie tej nie powinno się ręcznie wykonywać żadnych operacji (z doświadczenia dydaktycznego wiem, że utworzenie dodatkowych obiektów, co często przytrafia się studentom, nie powoduje żadnych skutków ubocznych, natomiast usunięcie obiektu systemowego może prowadzić do niepożądanego zachowania serwera, aż do całkowitej utraty możliwości posługiwania się nim);

- *msdb* — jest bazą wykorzystywaną podczas pracy serwisu *SQL Server Agent* do zarządzania zadaniami, alertami, pocztą, systemem powiadomień; również tej bazy dotyczą uwagi odnoszące się do ręcznej ingerencji użytkownika;
- *tempdb* — jest bazą przeznaczoną na obiekty tymczasowe, między innymi: pośrednie stany sortowań, informacje o stanie kursorów, lokalne i globalne tabele tymczasowe, stany pośrednie przed zatwierdzeniem transakcji; tworzenie przez użytkownika obiektów w tej bazie jest bezcelowe, ponieważ nie będą one utrwalone;
- *model* — jest bazą szablonem; wszystkie zawarte w tej bazie obiekty są przepisywane do każdej nowo tworzonej bazy danych, dlatego opłaca się w niej tworzyć obiekty, które będą wykorzystywane przez wiele baz, np. tabele słownikowe, procedury i funkcje wykorzystywane w każdej z baz (walidacja danych); należy pamiętać, że nie jest to narzędzie typu CASE, to znaczy, że utworzone w tej bazie nowe obiekty nie będą się automatycznie przenosić do już istniejących baz i operację taką należy wykonać ręcznie.

Ponadto istnieje baza systemowa *resource*, przeznaczona tylko do odczytu i widziana jako element schematu *sys*; w drzewie hierarchicznym niewidoczna, a dostępna tylko za pośrednictwem bazy *master*. Dla potrzeb realizacji replikacji, czyli synchronizacji danych, może zostać utworzona baza *distributor*. Dodatkowo w obrębie baz systemowych dostępny jest folder zawierający migawki baz danych — *Database Snapshots*.

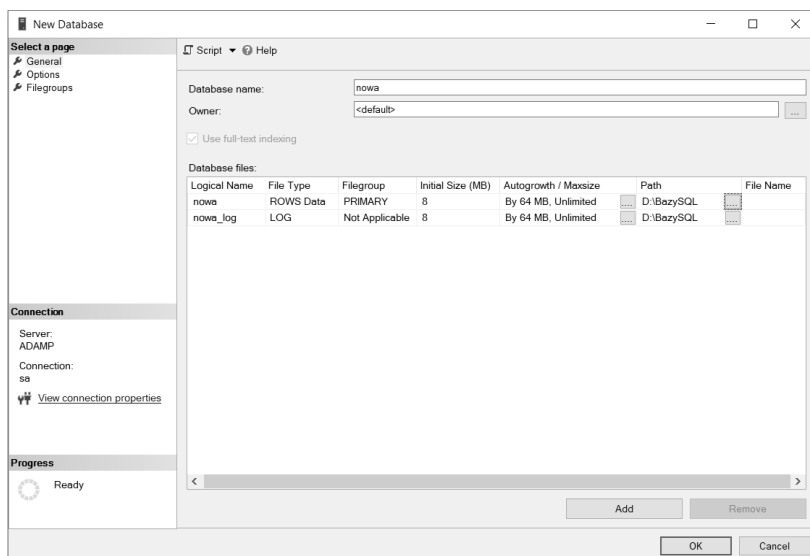
Poniżej baz systemowych widoczne są bazy treningowe dostarczone przez producenta (przy instalacji domyślnej bazy te nie są dodawane) oraz bazy danych utworzone przez użytkowników. Jeśli zostanie zainstalowany komponent *Reporting Services*, to dla potrzeb składowania definicji i przetwarzania raportów tworzone są bazy *ReportServer* (elementy trwałe) i *ReportServerTempDB* (obiekty nietrwałe, tymczasowe).

Aby utworzyć nową bazę danych z wykorzystaniem narzędzi wizualnych, należy prawym przyciskiem myszy kliknąć na poziomie węzła *Databases*. Na skutek takiego działania pojawi się menu kontekstowe, którego pierwszą pozycją jest *New Database...* — rysunek 2.17.



RYСУNEK 2.17. Wizualne tworzenie nowej bazy w Microsoft SQL Server Management Studio

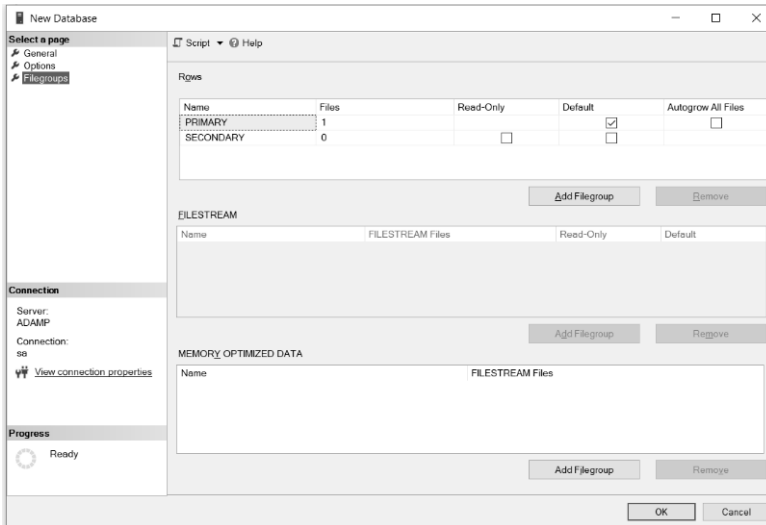
Po wybraniu tej pozycji pojawia się nowe okno dialogowe — rysunek 2.18. Na zakładce *General* konieczne jest podanie nazwy logicznej nowo tworzonej bazy danych, np. *nowa*. Powoduje to zdefiniowanie dwóch zbiorów fizycznych. Pierwszy z nich, o domyślnej nazwie pochodzącej od nazwy logicznej *nowa.mdf*, jest plikiem danych i zawiera wszystkie obiekty, które w bazie zostaną utworzone, a także dane zawarte w tabelach. Domyślnie ma on rozmiar 8 MB i jest automatycznie rozszerzany bez wskazania maksymalnego dopuszczalnego rozmiaru. Możliwe jest również wskazanie lokalizacji, w której zostanie utworzony. Drugi z nich, również o nazwie wywodzącej się z nazwy logicznej *nowa_log.ldf*, jest plikiem dziennika, przechowuje informacje o operacjach, jakie na bazie zostały wykonane, i jest wykorzystywany w procesie odtwarzania danych po wystąpieniu awarii. Ma domyślny rozmiar 8 MB i jest również automatycznie powiększany. Nazwy obu plików mogą być dowolnie zmienione przez użytkownika. Używając przycisku *Add*, możemy dodać kolejne pliki danych. Ponieważ najwolniejsze operacje wykonywane przez komputer to zawsze odczyt i zapis na nośniku fizycznym (dysku), dołożenie kolejnych plików danych umieszczonych na różnych dyskach spowoduje zrównoleglenie tych operacji (przynajmniej częściowe), co poprawia wydajność. Takie postępowanie nazywa się partycjonowaniem fizycznym i ma sens tylko dla dużych baz danych [8] [9] [10].



RYСУNEK 2.18. Wizualne tworzenie nowej bazy w Microsoft SQL Server Management Studio — zakładka *General*

Jeśli utworzymy wiele plików danych, to domyślnie będą się one znajdować w domyślnej grupie plików *PRIMARY*. Na zakładce *Filegroups* (rysunek 2.19) możliwe jest utworzenie kolejnych grup plików. W momencie utworzenia grupa jest pusta. Możliwe jest ustalenie dowolnej z nich jako grupy domyślnej, a w przypadku grup plików nieposiadających tej cechy możliwe

jest ustawienie właściwości tylko do odczytu (w takiej grupie nie można tworzyć żadnych nowych obiektów). Grupy plików mogą być usuwane (z wyjątkiem domyślnej), a przypisane do niej pliki fizyczne zostaną przeniesione do grupy domyślnej.

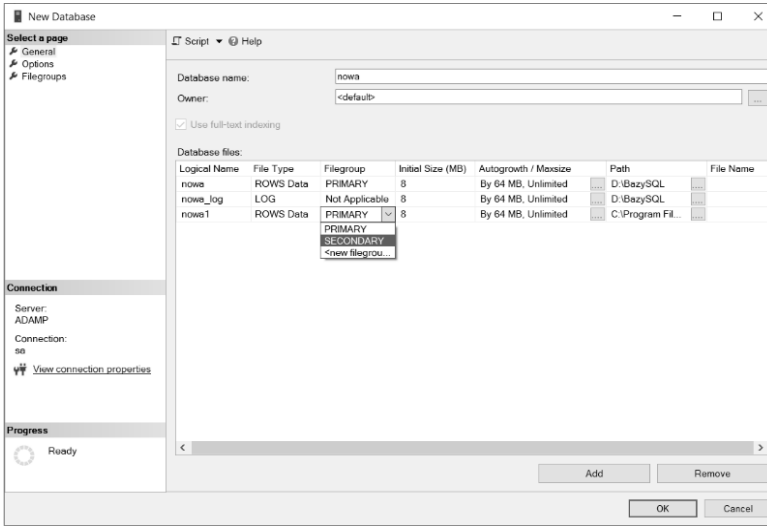


RYСУNEK 2.19. Wizualne tworzenie nowej bazy w Microsoft SQL Server Management Studio — zakładka Filegroups

Jeśli utworzymy wiele plików danych w domyślnej grupie plików, o tym, do którego z nich trafią nowo tworzone obiekty, np. tabele, decyduje silnik bazy danych, opierając się na wewnętrznych algorytmach równoważenia obciążenia [11]. Jeśli przypiszemy plik do grupy plików (rysunek 2.20), możemy podczas tworzenia zdecydować, do której z nich tworzony obiekt trafi, poprzez dopisanie klauzuli *ON NazwaGrupy* na końcu zapytania, np. *ON SECONDARY*. Jeśli jednak do tej grupy należy więcej plików niż jeden, znów o przydziale wewnątrz niej decyduje silnik bazy danych.

Poza dwoma poprzednio omawianymi zakładkami dostępna jest jeszcze zakładka *Options* — rysunek 2.21. Służy ona do ustawiania zaawansowanych właściwości bazy. Pozwala między innymi na ustawienie trybu zgodności z wersją SQL Server oraz trybu odzyskiwania po awarii. Pozostałe parametry są związane z automatyzacją procesów, domyślnymi ustawieniami kursorów, trybem zgodności z ANSI, ustawieniami przetwarzania, procesem powiadamiania oraz statusem bazy. W większości zastosowań ustawienia domyślne będą odpowiednie.

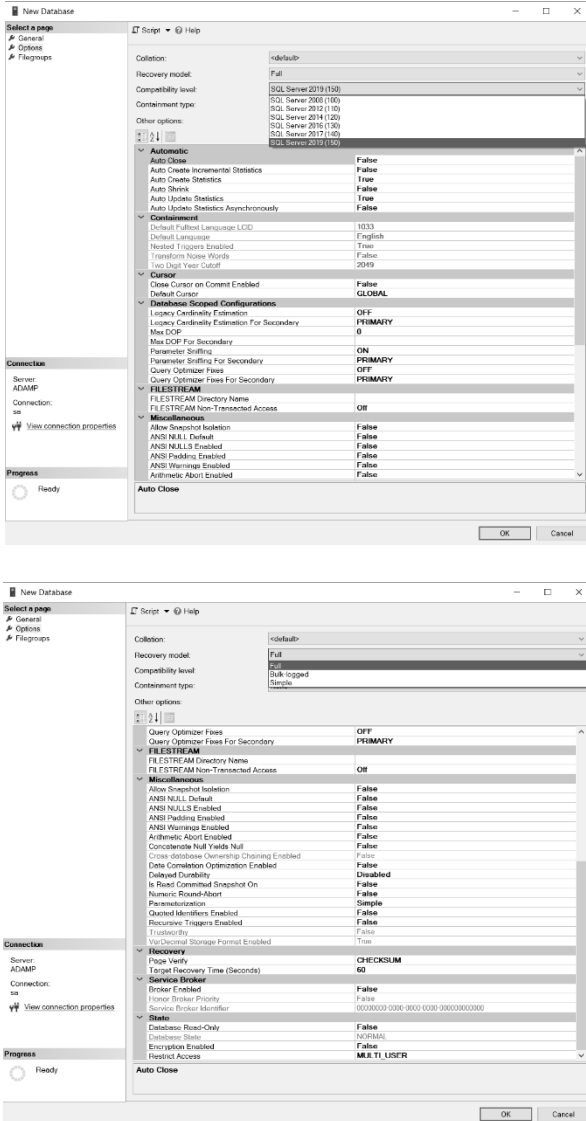
Gdy zostaną zatwierdzone dane ustawione na wszystkich trzech zakładkach okna dialogowego (rysunek 2.19), powstanie nowa baza danych. Będzie ona widoczna jako ostatnia pozycja w strukturze hierarchicznej. Po odświeżeniu widoku serwera pojawi się już w miejscu wynikającym z porządku alfabetycznego.



RYСУNEK 2.20. Wizualne tworzenie nowej bazy w Microsoft SQL Server Management Studio — zakładka General z uwzględnieniem dodanej grupy plików

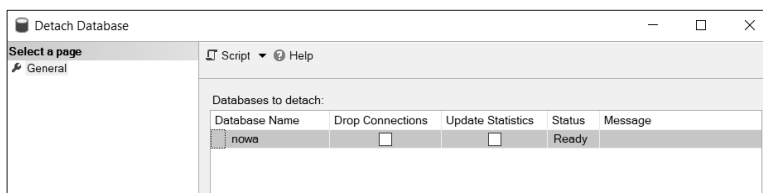
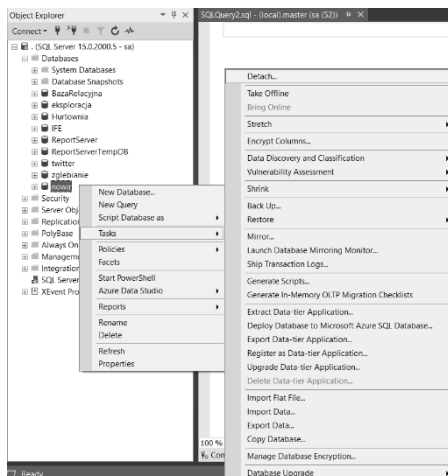
Utworzona na serwerze baza danych może zostać przeniesiona w dowolną lokalizację, również na inny komputer, aby mogła być obsługiwana za pomocą innego serwera SQL. Jednak dopóki jest dołączona do struktury logicznej baz obsługiwanych przez serwer, nie jest to możliwe, ponieważ zablokowane są wszystkie operacje dyskowe na plikach bazy oraz dziennika. Aby umożliwić migrację bazy, należy ją najpierw odłączyć, wybierając z menu kontekstowego dla bazy danych (kliknięcie prawym przyciskiem myszy) pozycję *Tasks*, w której z kolei wybieramy polecenie *Detach...* („odłącz”) (rysunek 2.22). W odpowiedzi pojawia się okno dialogowe zawierające informacje o statusie bazy danych oraz pozwalające na wybranie opcji aktualizującej statystyki bazy oraz powodującej usunięcie wszystkich aktywnych połączeń z bazą. Kliknięcie przycisku OK powoduje rozpoczęcie procesu odłączania. Jeśli z jakichś powodów proces ten nie powiedzie się, informacje o nich będzie można odczytać w pozycji *Message* tego samego okna dialogowego. Najczęstszą przyczyną powstawania błędów podczas odłączania jest występowanie aktywnych połączeń z bazą. Błąd ten powinien zostać usunięty w przypadku zaznaczenia opcji *Drop Connections*.

Jeśli uda nam się odłączyć bazę od struktury logicznej instancji serwera, możliwe będzie wykonanie operacji na jej plikach, np. przeniesienia ich w inne miejsce — na inny komputer. Powinniśmy taką operację przeprowadzać zarówno na pliku danych (*.mdf), jak i pliku dziennika (*.ldf). Po przeniesieniu w inne miejsce możemy bazę z powrotem przyłączyć do serwera (tego samego lub innego). W tym celu z menu kontekstowego dla węzła drzewa *Databases* wybieramy pozycję *Attach...* (rysunek 2.23). W rezultacie pojawia się okno dialogowe, w którym za pomocą przycisku *Add...* wskazujemy plik dołączanej bazy danych (*.mdf). Na skutek tego pojawia się informacja o dołączanej bazie: lokalizacja



RYSUNEK 2.21. Wizualne tworzenie nowej bazy w Microsoft SQL Server Management Studio — zakładka Options

pliku, pierwotna nazwa logiczna, nowa nazwa logiczna (domyślnie taka sama jak stara) oraz właściciel bazy (dbo — *database owner*, „właściciel bazy”). Ponadto dociągany jest plik dziennika (*.ldf) wraz z pozostałymi plikami danych, o ile istnieją. Informacja o plikach składających się na bazę jest zawarta w dolnej części okna dialogowego. Po potwierdzeniu wyboru rozpoczyna się proces dołączania. Ewentualne błędy pojawią się w pozycji *Message* tego samego okna.



RYSUNEK 2.22. Odłączanie bazy w Microsoft SQL Server Management Studio

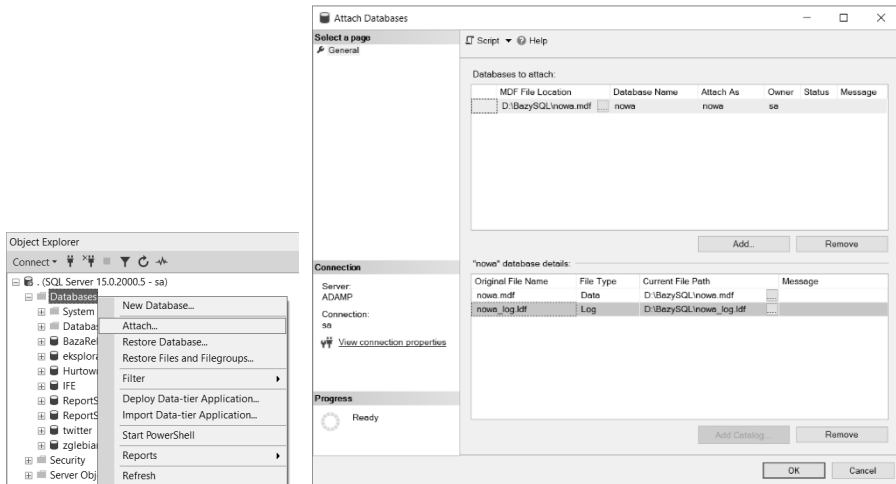
Jedną z podstawowych przyczyn takiego stanu jest uszkodzenie któregoś z plików lub brak pliku dziennika. Dołączenie bazy bez pliku dziennika przy użyciu narzędzi wizualnych nie jest kłopotliwe. Należy usunąć plik dziennika z okienka ...*database details* (aktywny na rysunku 2.22 przycisk *Remove*), a następnie wykonać dołączenie, co powoduje wygenerowanie pustego pliku dziennika. Dlatego należy bardzo dbać zarówno o plik dziennika, jak i plik danych. W przypadku uszkodzenia lub zagubienia pliku dziennika dołączenie bazy danych jest możliwe także za pomocą procedury systemowej:

```
sp_attach_db [@dbname = ] 'NazwaLogiczna',
[@filename1 = ] 'NazwaPlikuBazyDanych' [ ,...16 ]
```

Obowiązkowe jest podanie jednego zbioru danych. Należy jednak pamiętać, że jeśli nie zostanie podany plik dziennika, to będzie utworzony jego pusty zamiennik, co może powodować problemy z odzyskiwaniem danych po awarii, gdy konieczne jest odwołanie się do czasu sprzed utworzenia tego pliku. Możliwe jest również użycie analogicznej procedury:

```
sp_attach_single_file_db [@dbname = ] 'NazwaLogiczna',
[@physname = ] 'NazwaPlikuBazyDanych'
```

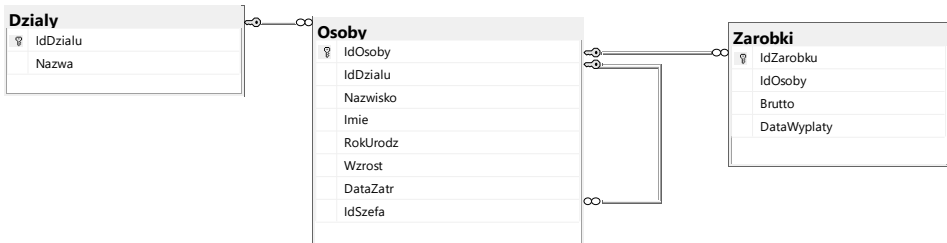
Jest ona dedykowana dla baz danych składających się z tylko jednego pliku danych. Przedstawiony mechanizm odłączania i dołączania bazy danych może być stosowany do migrowania danych na inne serwery oraz jako narzędzie wspierające tworzenie kopii zapasowych.



RYСУNEK 2.23. Dołączanie bazy w Microsoft SQL Server Management Studio

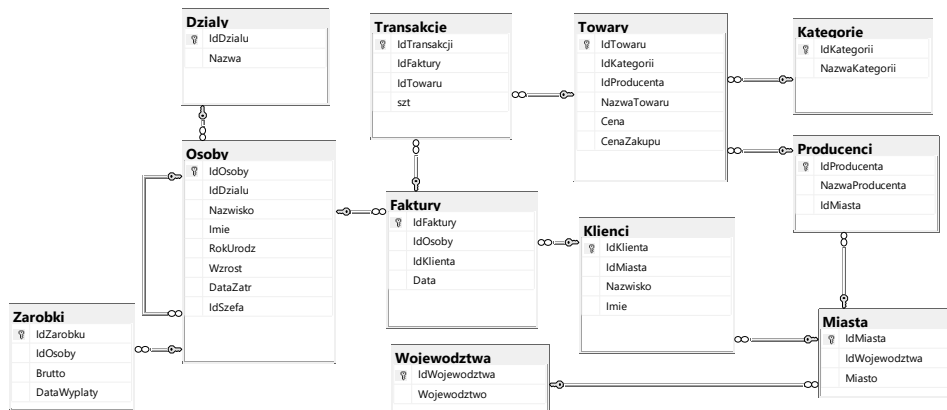
Omawiane mechanizmy powinny pozwolić na skuteczne zainstalowanie i skonfigurowanie MS SQL Server oraz na skorzystanie z przykładowej bazy danych dostępnej na stronie wydawnictwa, którą to bazę należy dołączyć do własnego serwera.

Zasadniczym elementem bazy danych jest struktura składająca się z trzech tabel pokazanych w postaci diagramu relacyjnego na rysunku 2.24. Przedstawia on schemat zatrudnienia w małej firmie. Na strukturę organizacyjną składają się działy (tabela *Działy*). Każdy pracownik (tabela *Osoby*) może być przypisany do najwyżej jednego działu. Pracownik może mieć najwyżej jednego przełożonego (*IdSzefa*), który jest również pracownikiem firmy. Pracownik otrzymuje wypłaty (tabela *Zarobki*), każda z nich jest przypisana do pracownika. Przez wypłatę rozumie się każdą operację finansową wykonaną na rzecz pracownika — wynagrodzenie za kolejne miesiące, premie, diety itp. Ten fragment schematu będzie wykorzystywany do omówienia większości przykładów zawartych w książce.



RYСУNEK 2.24. Diagram zasadniczej części schematu relacyjnego przykładowej bazy danych

Opisane poprzednio tabele stanowią fragment większego schematu, pokazanego na rysunku 2.25. Przedstawia on proces sprzedaży realizowany przez firmę (hurtownia, sklep). Firma dysponuje towarami (tabela *Towar*), które pogrupowane są w kategorie (tabela *Kategorie*).



RYСУNEK 2.25. Pełny diagram przykładowej bazy danych

Towary te były wyprodukowane przez firmy (tabela *Producenci*), które mają swoje siedziby w miastach (tabela *Miasta*), które znajdują się w województwach (tabela *Wojewodztwa*). Towary są kupowane przez klientów (tabela *Klienci*), którzy pochodzą z miast (tabela *Miasta*) znajdujących się w województwach (tabela *Wojewodztwa*). Dwie ostatnie tabele są wspólnymi słownikami opisującymi lokalizację zarówno producentów, jak i klientów. Na zakupione towary klienci mają wystawiane przez pracowników faktury (tabela *Faktury*), na których odnotowany jest fakt zakupu każdego z nich (tabela *Transakcje*).

Omówiony schemat jest znany Czytelnikom moich poprzednich książek [1], [2], [5], ale uważam go za wystarczająco ogólny, by wykorzystać go do omawiania większości zagadnień występujących w bazach danych, niezależnie od platformy czy rodzaju przetwarzania (transakcyjne, analityczne). Ponadto wydaje mi się dość prosty (zwłaszcza jego podstawowa część), tak że jego zrozumienie nie wymaga specjalnego, dodatkowego wysiłku, co daje więcej czasu na przyswojenie sobie prezentowanych przykładów. W przyjętym nazewnictwie z premedytacją nie stosowano narodowych znaków diakrytycznych, co jest dopuszczalne, ale komplikuje później pisanie zapytań. Nazwy kluczy głównych zaczynają się od prefiksu *Id*, po którym następuje rzeczownik w liczbie pojedynczej określający zawartość tabeli lub tabeli nadrzędnej. Oznacza to, że pola kluczy głównych oraz kluczy obcych mają te same nazwy. Jedynym wyjątkiem jest pole klucza wewnętrznego *IdSzefa*, gdzie zastosowano ten sam prefiks, ale aby nie doprowadzić do dublowania nazw kolumn w tabeli, co jest zabronione, ciąg dalszy nie określa nazwy tabeli, lecz jedynie wskazuje na rolę pola.

PROGRAM PARTNERSKI

— GRUPY HELION —

- 
1. ZAREJESTRUJ SIĘ
 2. PREZENTUJ KSIĄŻKI
 3. ZBIERAJ PROWIZJĘ

Zmień swoją stronę WWW w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!

<http://program-partnerski.helion.pl>

GRUPA
Helion

Skorzystaj z potężnego wsparcia w zarządzaniu danymi

Procesy gromadzenia danych, zarządzania nimi, przetwarzania ich i izolowania ze zbioru tych, które są aktualnie potrzebne, to wielkie wyzwanie XXI wieku. Dane to wiedza, wiedza to potęga – przy czym samo posiadanie wiedzy nie czyni jeszcze potężnym, trzeba być w stanie nią zarządzać dla swoich potrzeb. W firmie oznacza to na przykład sprawne poruszanie się w gąszczu zamówień klientów, zasobach magazynu, przepływach towarów i łańcuchach dostaw.

By wspomóc procesy związane z zarządzaniem danymi, środowisko informatyczne posługuje się systemami, które pozwalają nimi zarządzać – takimi jak MS SQL Server, wspierany i rozpowszechniany przez firmę Microsoft. System ten słynie z szerokiej możliwości technologicznych, dużej elastyczności i wydajności, ma więc wszystko, czego potrzeba do zarządzania nawet najbardziej rozbudowanymi bazami danych. By rozpocząć pracę z zasobami, które oferuje MS SQL Server, trzeba poznać zasady, na jakich działa.

- **Dowiedz się, jak zainstalować i skonfigurować środowisko MS SQL Server**
- **Poznaj język zapytań SQL dla systemu MS SQL Server**
- **Przyjrzyj się problemom, jakie rozwiązuje SQL**
- **Zapoznaj się z rozszerzeniem proceduralnym Transact-SQL**
- **Opanuj zasady przetwarzania transakcyjnego**
- **Zrozum typy złożone**
- **Naucz się rozwiązywać problemy za pomocą rozszerzeń proceduralnych i obiektowych**

	<p><i>Sprawdź nasze szkolenia!</i></p>  <p>SZKOLENIA</p> <p>AKADEMIA IT & BUSINESS</p> <p>HELIONSZKOLENIA.PL</p>	<p>KOD KORZYŚCI <i>Sięgnij po więcej!</i></p> 
 helion.pl		
 HELION SA ul. Kościuszki 1c 44-100 Gliwice tel.: 32 230 98 63 helion@helion.pl		
INFORMATYKA W NAJLEPSZYM WYDANIU		<p>ISBN 978-83-283-7412-6</p>  <p>9 788328 374126</p> <p>Cena: 119,00 zł</p>