

Leonard G. Lobel
Eric D. Boyd

Microsoft® Azure™ SQL Database *Krok po kroku*

Przekład: Marek Włodarz

APN Promise, Warszawa 2014

Microsoft® Azure™ SQL Database Krok po kroku

Authorized Polish translation of the English language edition entitled: Microsoft® Azure™ SQL Database Step by Step, ISBN: 978-0-7356-7942-9 by Leonard G. Lobel and Eric D. Boyd, published by Pearson Education, Inc, publishing as Microsoft Press, A Division Of Microsoft Corporation.

Copyright © 2014 by Leonard G. Lobel and Eric D. Boyd

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

Polish language edition published by APN PROMISE SA Copyright © 2014

Autoryzowany przekład z wydania w języku angielskim, zatytułowanego: Microsoft® Azure™ SQL Database Step by Step, ISBN: 978-0-7356-7942-9 by Leonard G. Lobel and Eric D. Boyd, opublikowanego przez Pearson Education, Inc, publikującego jako Microsoft Press, oddział Microsoft Corporation.

Wszystkie prawa zastrzeżone. Żadna część niniejszej książki nie może być powielana ani rozpowszechniana w jakiegokolwiek formie i w jakikolwiek sposób (elektroniczny, mechaniczny), włącznie z fotokopiowaniem, nagrywaniem na taśmy lub przy użyciu innych systemów bez pisemnej zgody wydawcy.

APN PROMISE SA, biuro: ul. Kryniczna 2, 03-934 Warszawa
tel. +48 22 35 51 600, fax +48 22 35 51 699
e-mail: mspress@promise.pl

Książka ta przedstawia poglądy i opinie autorów. Przykłady firm, produktów, osób i wydarzeń opisane w niniejszej książce są fikcyjne i nie odnoszą się do żadnych konkretnych firm, produktów, osób i wydarzeń, chyba że zostanie jednoznacznie stwierdzone, że jest inaczej. Ewentualne podobieństwo do jakiegokolwiek rzeczywistej firmy, organizacji, produktu, nazwy domeny, adresu poczty elektronicznej, logo, osoby, miejsca lub zdarzenia jest przypadkowe i niezamierzone.

Nazwa Microsoft oraz znaki towarowe wymienione na stronie <http://www.microsoft.com/about/legal/en/us/IntellectualProperty/Trademarks/EN-US.aspx> są zastrzeżonymi znakami towarowymi grupy Microsoft. Wszystkie inne znaki towarowe są własnością ich odnośnych właścicieli.

APN PROMISE SA dołożyła wszelkich starań, aby zapewnić najwyższą jakość tej publikacji. Jednakże nikomu nie udziela się rękojmi ani gwarancji.

APN PROMISE SA nie jest w żadnym wypadku odpowiedzialna za jakiegokolwiek szkody będące następstwem korzystania z informacji zawartych w niniejszej publikacji, nawet jeśli APN PROMISE została powiadomiona o możliwości wystąpienia szkód.

ISBN: 978-83-7541-153-9

Przekład: Marek Włodarz
Korekta: Ewa Swędrowska
Skład i łamanie: MAWart Marek Włodarz

Spis treści

<i>Wprowadzenie</i>	ix
1 Poznajemy Microsoft Azure SQL Database	1
Przetwarzanie w chmurze: przedstawienie pojęcia	2
Natychmiastowe, dynamiczne wdrażanie	2
Chmura Microsoft Azure	3
Rejestrowanie się dla SQL Database	6
Tworzenie konta Microsoft	6
Tworzenie subskrypcji Microsoft Azure	8
Tworzenie serwera	10
Tworzenie instancji SQL Database	14
Korzystanie z portalu zarządzania SQL Database	16
Projektowanie tabel i relacji	18
Wstawianie danych	22
Odpytywanie bazy danych	26
Poznajowanie dodatkowych możliwości portalu	29
Podsumowanie	32
2 Konfiguracja i ceny	33
Korzystanie z portalu zarządzania platformą Microsoft Azure	33
Tworzenie nowej bazy danych	33
Ustawianie reguł zapory	37
Uzyskiwanie łańcucha połączenia	41
Usuwanie bazy danych	42
Korzystanie z SQL Server Management Studio	43
Łączenie się z SQL Database	44
Tworzenie nowej bazy danych	46
Zmianie wydania i maksymalnego rozmiaru bazy danych	47
Usuwanie bazy danych	47
Korzystanie z PowerShell	47
Instalowanie poleceńcmdlet dla Microsoft Azure	47
Korzystanie z PowerShell ISE	49
Konfigurowanie PowerShell dla konta Microsoft	50
Tworzenie nowego serwera	51
Tworzenie nowej bazy danych	52
Usuwanie bazy danych	53

Planowanie budżetu dla SQL Database	54
Magazyn SQL	54
Pasma klienckie	55
Magazyn dla kopii zapasowych.....	56
Pasma magazynu kopii zapasowych.....	57
Wsparcie	58
Optymalizacja kosztów	59
Konfigurowanie wydania i rozmiaru bazy danych	60
Podsumowanie	61
3 Różnice pomiędzy SQL Server a Microsoft Azure SQL Database	63
Ograniczenia rozmiaru	64
Ograniczenia połączeń	65
Nieobsługiwane funkcjonalności.....	65
Podsumowanie	69
4 Migrowanie baz danych	71
Przygotowania wstępne do migracji danych.....	72
Migrowanie danych za pomocą skryptów Transact-SQL	72
Konfigurowanie lokalnej bazy danych SQL Server	72
Wypełnianie bazy danych przy użyciu T-SQL	75
Generowanie skryptów T-SQL	76
Aplikacje warstwy danych SQL.....	79
Tworzenie konta w Microsoft Azure Storage.....	79
Eksportowanie BACPAC do Microsoft Azure Storage	83
Importowanie pliku BACPAC do Microsoft Azure SQL Database.....	86
SQL Server Bulk Copy (bcp)	90
Migrowanie schematu	90
Eksportowanie danych	93
Importowanie danych	94
SQL Database Migration Wizard	97
Pobieranie narzędzia	98
Migrowanie bazy danych	99
Podsumowanie	106
5 Bezpieczeństwo i kopie zapasowe.....	107
Główne obawy dotyczące chmury.....	108
Odpowiedzialność dostawcy chmury publicznej za bezpieczeństwo	108
Wspólna odpowiedzialność za zabezpieczenia.....	109
Bezpieczeństwo w Microsoft Azure.....	110

Zabezpieczanie SQL Database	111
Tworzenie bazy danych SQL Database	111
Konfigurowanie zapory SQL Database	112
Uwierzytelnianie i autoryzowanie użytkowników	117
Tworzenie kopii zapasowej SQL Database	124
Kopiowanie bazy danych	125
Monitorowanie postępów operacji kopiowania bazy danych	126
Eksportowanie pliku BACPAC	128
Importowanie pliku BACPAC	130
Tworzenie harmonogramu eksportowania BACPAC	133
Podsumowanie	135
6 Raportowanie w chmurze	137
Tworzenie maszyny wirtualnej dla SQL Server Reporting Services	139
Tworzenie maszyny wirtualnej z galerii obrazów	140
Konfigurowanie SSRS na maszynie wirtualnej	143
Otwieranie dostępu przez zaporę do serwera raportów	145
Tworzenie przykładowej bazy danych	148
Korzystanie z Report Builder	150
Instalowanie Report Builder	151
Tworzenie raportu przy użyciu Report Builder	152
Korzystanie z projektów Visual Studio Report Server	167
Instalowanie bazy danych AdventureWorks2012 w SQL Database	168
Instalowanie SSDT Business Intelligence dla Visual Studio 2012	171
Tworzenie raportu przy użyciu Visual Studio	173
Implementowanie zabezpieczeń raportu	188
Wyłączanie maszyny wirtualnej SSRS	189
Podsumowanie	189
7 Microsoft Azure SQL Data Sync	191
Poznanie SQL Data Sync	191
Eksportowanie danych z SQL Server do SQL Database	193
Importowanie danych z SQL Database do SQL Server	194
Współużytkowanie danych w wielu lokalizacjach	195
Skalowanie usług	196
Tworzenie bazy SQL Database	198

Korzystanie z SQL Data Sync	200
Tworzenie grupy synchronizacji.....	201
Tworzenie reguł synchronizacji	207
Wykonywanie ręcznej synchronizacji	210
Definiowanie rozwiązywania konfliktów.....	215
Tworzenie harmonogramu automatycznej synchronizacji	218
Tworzenie lokalnej bazy SQL Server	220
Tworzenie agenta synchronizacji.....	223
Potencjalne pułapki i najlepsze praktyki	234
Podsumowanie	235
8 Projektowanie i dostrajanie na potrzeby skalowalności i wysokiej wydajności	237
Osiąganie wysokiej wydajności w chmurze	238
Tworzenie API typu RESTful	239
Tworzenie przykładowej bazy danych	239
Tworzenie nowego rozwiązania.....	241
Tworzenie projektu ASP.NET.....	242
Dodawanie kodu pierwszego kontrolera Web API (typu Entity Framework)..	244
Testowanie Web API <i>Wine</i>	249
Dodawanie kontrolera Web API ADO.NET.....	251
Testowanie Web API <i>Customer</i>	254
Zarządzanie połączeniami SQL Database	254
Otwierać późno, zamykać wcześniej	254
Korzystanie z pul połączeń.....	255
Przywracanie po błędach połączeń.....	255
Dodawanie Transient Fault Handling Application Block	256
Stosowanie Transient Fault Handling Application Block w ADO.NET	258
Stosowanie Transient Fault Handling Application Block w Entity Framework.	260
Redukowanie opóźnień sieciowych	264
Umieszczanie usług blisko siebie.....	265
Minimalizowanie ponownych obiegów	265
Efektywne korzystanie z SQL Database	266
Używanie najlepszej usługi magazynowania.....	266
Optymalizowanie zapytań	267
Skalowanie SQL Database w górę	267
Partycjonowanie danych	272
Skalowanie przy użyciu partycji funkcjonalnych.....	273
Skalowanie przy użyciu partycjonowania horyzontalnego.....	273
Podsumowanie	282

9	Monitorowanie i zarządzanie SQL Database	283
	Tworzenie przykładowej bazy danych	283
	Monitorowanie	284
	Stosowanie portalu zarządzania	284
	Tablica kontrolna Microsoft Azure	289
	Portal zarządzania SQL Database	292
	Dynamiczne widoki i funkcje zarządzania	297
	Programowanie API typu REST do zarządzania usługą	303
	Podsumowanie	310
10	Budowanie rozwiązania w chmurze	311
	Tworzenie bazy danych SQL Database	314
	Rozbudowywanie bazy danych	316
	Tworzenie nowego rozwiązania	317
	Tworzenie projektu typu SQL Server Database	318
	Ustawianie platformy docelowej	319
	Importowanie bazy SQL Database do projektu	320
	Dodawanie nowej kolumny do tabeli <i>Wine</i>	323
	Wdrażanie zmodyfikowanego projektu z powrotem w Microsoft Azure SQL Database	325
	Tworzenie tabeli <i>Order</i>	328
	Tworzenie procedur składowanych dla tabeli <i>Order</i>	331
	Tworzenie warstwy dostępu do danych (DAL)	336
	Wprowadzenie do Entity Data Model	338
	Tworzenie projektu Data Access Layer	339
	Tworzenie Entity Data Model	340
	Tworzenie witryny Web	346
	Tworzenie projektu aplikacji Web ASP.NET	347
	Dodawanie referencji do warstwy dostępu do danych	348
	Tworzenie interfejsu użytkownika	350
	Lokalne testowanie witryny	354
	Wdrażanie witryny w Microsoft Azure	358
	Tworzenie usług Web Api ASP.NET	362
	Dodawanie kontrolera Web API	363
	Testowanie Web API	366
	Wdrażanie Web API w chmurze	367

Tworzenie aplikacji Windows Phone	368
Instalowanie Windows Phone SDK 8.0	369
Tworzenie projektu typu Windows Phone	370
Dodawanie Json.NET	371
Tworzenie głównej strony aplikacji	372
Testowanie aplikacji Windows Phone	381
Podsumowanie	383
<i>Indeks</i>	385
<i>O autorach</i>	402

Wprowadzenie

Microsoft Azure SQL Database to działająca w chmurze wersja Microsoft SQL Server, czyli doskonale znanej platformy relacyjnej bazy danych. Niezależnie od pewnych wartych odnotowania różnic, SQL Database (krótsza nazwa Microsoft Azure SQL Database) jest w większości kompatybilna z SQL Server, zatem doświadczenie już zdobyte przy pracy z SQL Server może być w znacznym stopniu bezpośrednio zastosowane w SQL Database. Jeśli jesteś profesjonalistą od oprogramowania, rozważającego wykorzystanie chmury jako platformy dla bazy danych w swojej kolejnej aplikacji, SQL Database może być właśnie tym narzędziem, którego potrzebujesz. A jeśli chcesz szybko dołączyć do tej rozwijającej się platformy, bez względu na to, czy masz już jakieś doświadczenia z SQL Server, czy nie, ta książka na pewno w tym pomoże.

Microsoft Azure SQL Database krok po kroku proponuje uporządkowaną wycieczkę po platformie SQL Database. Naszym celem było pokazanie SQL Database od początku do końca z zachowaniem rozsądnej równowagi pomiędzy zakresem tematyki i zagłębianiem się w szczegóły. W pierwszym rozdziale szybko utworzymy pierwszą bazę danych w Microsoft Azure. Do końca ostatniego rozdziału zbudujemy pełne, wielowarstwowe rozwiązanie bazodanowe w chmurze – obejmujące witrynę Web i aplikację dla Windows Phone 8 – w pełni oparte na SQL Database. W każdym z rozdziałów będziemy poznawali inne oblicze SQL Database i wiele z towarzyszących jej technologii. SQL Database to rozległy temat, ale staraliśmy się tak skonstruować każdy rozdział, aby zajmować się tylko jedną rzeczą na raz i pokazywać łatwe do naśladowania procedury, które pozwalają przenieść wyrefinowane koncepcje na bezpośrednie zastosowania praktyczne. Nasza wiedza będzie rosła w każdym rozdziale w miarę poznawania zagadnień konfigurowania, migracji, zabezpieczeń, kopii zapasowych i wielu innych.

Wielka różnica pomiędzy oprogramowaniem działającym lokalnie (w siedzibie firmy – *on-premises*) a usługami w chmurze polega na tym, że tę drugą można aktualizować i rozbudowywać znacznie częściej i prościej, niż pierwszą, jako że w przypadku chmury nie jest wymagana żadna instalacja czy tworzenie infrastruktury klienckiej. Z drugiej strony usługi w chmurze podlegają również częstym zmianom, jeśli chodzi o ich cenę. Z tych względów funkcje, ograniczenia, koszty, interfejs użytkownika, a nawet sama nazwa Microsoft Azure SQL Database, opisane w tej książce, mogą zmienić się do czasu, gdy książka trafi do rąk Czytelnika. Na przykład, krótko przed skierowaniem książki do druku platforma nazywana wcześniej Windows Azure została przemianowana na Microsoft Azure. (Choć tytuł książki i tekstowe odsyłacze zostały odpowiednio zaktualizowane, wiele zrzutów ekranowych nadal pokazuje starszą

nazwę *Windows Azure*). Jednak niezależnie od potencjalnych zmian tego rodzaju zasady i techniki omawiane w książce pozwolą osiągnąć poziom sprawnego posługiwania się Microsoft Azure SQL Database.



UWAGA W miarę ewolucji platformy Azure zmieniamy się wraz z nią. Choć pierwsze wydanie dopiero zostało wysłane do druku, pracujemy intensywnie nad następnym wydaniem z rozszerzonym omówieniem nowo przedstawionych (zapowiedzianych) wydań Basic, Standard i Premium. Te nowe warstwy usługi (których dostępność była jeszcze ograniczona do wersji testowej w chwili pisania tych słów) pozwalają obsłużyć większe i bardziej skalowalne bazy danych, niż oferowane przez obecne wydania Web i Business. Kolejne wydanie zostanie również uzupełnione o nowy portal zarządzania, który jest obecnie opracowywany przez firmę Microsoft.

Kto powinien przeczytać tę książkę

Książka ta ma pomóc deweloperom oprogramowania i profesjonalistom bazodanowym zrozumieć podstawowe koncepcje Microsoft Azure SQL Database i powiązane z nią technologie. Czytelnicy mogą nie mieć żadnego lub tylko niewielkie doświadczenie w pracy z SQL Server i nie stanowi to żadnego problemu. Naszą wycieczkę rozpoczynamy od samego początku i zapewniamy dobry punkt startowy, nawet jeśli Czytelnik nie dysponuje żadną wiedzą o SQL Server lub koncepcjach relacyjnych baz danych.

Książka będzie też użyteczna dla osób, które dobrze znają lokalne instalacje SQL Server i są zainteresowane tworzeniem nowych aplikacji wykorzystujących SQL Database lub tych, którzy chcą przenieść istniejące rozwiązania działające lokalnie do chmury.

Założenia

Nie zakładamy ani nie wymagamy żadnego wcześniejszego doświadczenia ani wiedzy dotyczącej Microsoft Azure (i w ogólności – przetwarzania w chmurze). Co więcej, choć znajomość Microsoft SQL Server na pewno jest przydatna, również nie jest wymagana.

Kilka rozdziałów odwołuje się do programowania na platformie .NET. Także w tym przypadku wcześniejsze doświadczenia w pracy z Microsoft Visual Studio i językiem C# będzie pomocne, ale niekonieczne. Procedury zawarte w tych rozdziałach zawierają cały potrzebny kod wraz z jasnym wyjaśnieniem jego działania.

Ta książka może nie być odpowiednia, jeśli...

Książka ta nie jest przeznaczona dla osób, które już mają rozległą wiedzę i doświadczenie w pracy z SQL Database i szukają pogłębionych informacji o wewnętrznych mechanizmach lub innych specjalistycznych zagadnieniach, które nie zostały przedstawione w tej książce. Tym niemniej książka nadal zawiera informacje, które mogą się okazać użyteczne także dla doświadczonych użytkowników. Z tego względu sugerujemy przejrzeć omówienia poszczególnych rozdziałów w kolejnym podrozdziale.

Organizacja książki

Książka zawiera dziesięć rozdziałów, w których każdy skupia się na innym aspekcie Microsoft Azure SQL Database. Większość Czytelników powinna rozpocząć od rozdziału 1 i posuwać się dalej konsekwentnie, ale można też sobie wyobrazić czytanie w dowolnie wybranej kolejności. W każdym przypadku Czytelnik znajdzie praktyczne wskazówki i sposoby postępowania, które pomogą wykonać pracę z SQL Database.

- **Rozdział 1 – Poznajemy Microsoft Azure SQL Database** Otwierający rozdział zawiera ogólny przegląd platformy SQL Database. Po krótkim omówieniu przetwarzania w chmurze utworzymy konto Microsoft (o ile Czytelnik jeszcze go nie ma) i próbną (bezpłatną) subskrypcję Microsoft Azure. Następnie nauczymy się, jak posługiwać się portalem zarządzania Microsoft Azure do tworzenia serwerów i baz danych. Później przejdziemy do korzystania z portalu zarządzania SQL Database, w którym będziemy projektować tabele, widoki i procedury składowane, a następnie wypełniać tabele przykładowymi danymi i wykonywać zapytania. Żadne narzędzia lokalne nie są wymagane, aby przejść wszystkie procedury zawarte w tym rozdziale; wszystko, czego potrzebujemy, to przeglądarka sieci Web i dostęp do Internetu.
- **Rozdział 2 – Konfiguracja i ceny** Po przedstawieniu podstaw ten rozdział wyjaśnia dodatkowe opcje konfigurowania SQL Database, wykraczające poza stosowanie przeglądarki i portali, omówione w rozdziale 1. Nauczymy się, jak połączyć się z SQL Database przy użyciu standardowych narzędzi lokalnych, takich jak SQL Server Management Studio (SSMS) oraz SQL Server Data Tools (SSDT), stanowiące część Visual Studio. Dowiemy się też, jak konfigurować i zarządzać SQL Database przy użyciu powłoki PowerShell, wykorzystując moduł poleceń cmdlet Microsoft Azure dla PowerShell. Rozdział kończy omówienie struktury kosztów dotyczących SQL Database i wskazówki pomagające w zaplanowaniu budżetu dla rozwiązania SQL Database.
- **Rozdział 3 – Różnice pomiędzy SQL Server a Microsoft Azure SQL Database** Czytelnicy dysponujący doświadczeniem w pracy z SQL Server będą chcieli poznać ważne różnice pomiędzy silnikiem relacyjnej bazy danych działającym w siedzibie firmy a implementacją SQL Database w chmurze Microsoft Azure. Ten

krótki rozdział zawiera wyliczenie tych różnic i wyjaśnia ich przyczyny. Tam, gdzie to możliwe, zasugerowaliśmy również obejścia dla funkcji SQL Server, które nie są wspierane w SQL Database.

- **Rozdział 4 – Migrowanie baz danych** Przy budowaniu systemu mającego działać w chmurze Microsoft Azure często zachodzi potrzeba przeniesienia baz danych z istniejących serwerów SQL Server do SQL Database. Istnieje wiele technik i narzędzi, których można użyć w celu przemieszczenia baz danych SQL Database. W tym rozdziale pokażemy wykorzystanie do tego celu skryptów Transact-SQL, mechanizmu SQL Data-Tier Applications, kopiowania masowego oraz kreatora SQL Database Migration Wizard.
- **Rozdział 5 – Bezpieczeństwo i kopie zapasowe** Bezpieczeństwo, dostępność i możliwość przywrócenia po awarii to czołowe pozycje na liście rzeczy, które klienci rozważają zastanawiając się nad korzystaniem z usług publicznej chmury. W tym rozdziale poznamy mechanizmy zabezpieczeń dostępne w Microsoft Azure. Pokażemy, jak zabezpieczyć bazę danych SQL Database przy użyciu reguł zapory, a także jak tworzyć użytkowników i przypisywać uprawnienia. Na koniec omówimy sposoby przywracania po awariach przy użyciu technik kopii zapasowych udostępnianych przez SQL Database.
- **Rozdział 6 – Raportowanie w chmurze** Jeśli mamy już bazę danych zawierającą jakieś informacje, jest tylko kwestią czasu, gdy pojawi się wymóg tworzenia raportów dotyczących tych danych. Gdy baza ta jest hostowana w chmurze Microsoft Azure, naturalne jest rozważenie umieszczenia w tej chmurze również rozwiązania raportującego. W tym rozdziale dowiemy się, jak utworzyć maszynę wirtualną (VM) w Azure, hostującą SQL Server Reporting Services (SSRS) w chmurze (nie jest wymagane wcześniejsze doświadczenie w pracy z SSRS). Po skonfigurowaniu VM pokażemy, jak budować raporty SSRS przy użyciu dwóch narzędzi autorskich: Report Builder oraz SSDT Business Intelligence dla Visual Studio. Po zbudowaniu i przejrzaniu raportów lokalnie nauczymy się, jak wdrożyć je na maszynie wirtualnej, aby zbudować pełne rozwiązanie działające w chmurze.
- **Rozdział 7 – Microsoft Azure SQL Data Sync** W tym rozdziale nauczymy się, jak wykorzystać usługę SQL Data Sync dostępną w Microsoft Azure do replikowania danych pomiędzy wieloma bazami danych. Poznamy architekturę gwiazdy, na której oparta jest usługa, a następnie pokażemy, jak wykorzystać usługę SQL Data Sync do implementowania rozwiązań w rozmaitych scenariuszach, w tym jedno- i dwukierunkowe replikacje w zbiorze baz danych rozmieszczonych w różnych lokalizacjach. Procedury w tym rozdziale poprowadzą Czytelnika poprzez proces konfigurowania usługi SQL Data Sync i tworzenia grup synchronizacji obejmujących zarówno bazy danych hostowane w chmurze (w Microsoft Azure SQL Database), jak i w siedzibie (używające SQL Server). Nauczmy się też, jak konfigurować strategię rozwiązywania konfliktów i włączyć harmonogram automatycznej synchronizacji.

- **Rozdział 8 – Projektowanie i dostrajanie na potrzeby skalowalności i wysokiej wydajności** Aplikacje i systemy przeznaczone do rzeczywistych zastosowań produkcyjnych muszą zapewniać dobrą wydajność i niezawodność działania. W tym rozdziale zajmiemy się optymalizacją i dostrajaniem wydajności bazy danych w SQL Database. Następnie poprawimy stabilność aplikacji poprzez zarządzanie połączeniami bazy danych i obsługą błędów połączeń, wykorzystując zarówno ADO.NET, jak i platformę Entity Framework. Na koniec pokażemy, jak wykonać skalowanie bazy danych w SQL Database przy użyciu specjalnej techniki partycjonowania znanej jako *sharding*.
- **Rozdział 9 – Monitorowanie i zarządzanie SQL Database** Usługi używane w aplikacjach produkcyjnych muszą zapewniać możliwości monitorowania i zarządzania. W tym rozdziale dowiemy się, jak monitorować kondycję SQL Database przy użyciu portalu zarządzania, tablicy kontrolnej Microsoft Azure i dynamicznych widoków i funkcji zarządzania. Nauczymy się też jak programowo zautomatyzować operacje w SQL Database przy użyciu opartego na standardzie REST komponentu Service Management API.
- **Rozdział 10 – Budowanie rozwiązania w chmurze** W końcowym rozdziale książki pokażemy budowanie kompletnego rozwiązania w chmurze, opartego na Microsoft Azure SQL Database. Mówiąc konkretnie, zbudujemy rozwiązanie Visual Studio, zawierające projekt bazodanowy SQL Server Database, warstwę dostępu do danych Entity Framework, witrynę ASP.NET MVC oraz usługę Web API. Rozwiązanie to będzie tworzyło witrynę do składania zamówień oraz usługi Web i aplikację Windows Phone 8 pozwalającą na odczytywanie, dodawanie i modyfikowanie danych przechowywanych w SQL Database.

Konwencje stosowane w książce

Treści zawarte w książce wykorzystują kilka konwencji, które mają na celu zapewnić większą czytelność:

- Każda procedura składa się z serii zadań przedstawianych jako numerowane kroki (1, 2 i tak dalej) wyliczające wszystkie akcje, które trzeba podjąć, aby ukończyć ćwiczenie.
- Elementy w ramkach, takie jak „Uwaga” lub „Wskazówka”, zawierają dodatkowe informacje lub alternatywne metody realizacji danego zadania.
- Tekst, który ma zostać wpisany przez Czytelnika (poza blokami kodu), jest wytłuszczony.
- Nazwy poleceń, elementów menu i opcji, które należy kliknąć lub które po prostu identyfikują elementy interfejsu, są wyróżnione **czcionką bezszeryfową**. Przy pierwszym użyciu jakiegoś terminu w nawiasie podawane jest jego tłumaczenie na język polski.

- Nazwy elementów programowych, adresy internetowe oraz nowe lub ważne pojęcia prezentowane są *kursywą*.
- Znak plus (+) pomiędzy nazwami klawiszy oznacza, że klawisze te należy nacisnąć jednocześnie. Na przykład „Naciśnij Alt+Tab” oznacza, że klawisz Alt musi być przytrzymany, gdy naciskamy klawisz Tab.
- Pionowa kreska pomiędzy elementami menu (na przykład File | Close) oznacza, że należy wybrać pierwszy element menu, a następnie drugi z menu podrzędnego (i tak dalej, jeśli elementów jest więcej).

Wymagania systemowe

Zasadniczo nie istnieją specjalne wymagania systemowe do pracy z SQL Database. Portal Microsoft Azure wymaga jedynie kompatybilnej przeglądarki Web oraz dostępu do Internetu. Korzystanie z portalu zarządzania SQL Database wymaga przeglądarki z zainstalowaną wtyczką Silverlight.

Niektóre rozdziały zawierają procedury wykorzystujące narzędzia zainstalowane lokalnie – zazwyczaj SQL Server Management Studio (SSMS) oraz Visual Studio 2013. Aby wykonać te procedury, konieczne jest zainstalowanie tych programów, co oznacza spełnienie następujących wymagań

- System operacyjny Windows 7, Windows 8.*, Windows Server 2008 SP2, Windows Server 2008 R2 albo Windows Server 2012 (R2).
- Dowolne wydanie Visual Studio 2013 (przy korzystaniu z wydania Express konieczne może być pobranie wielu dodatkowych elementów).
- SQL Server 2012 Express Edition lub wyższa wersja, z komponentem SQL Server Management Studio 2012 Express lub wyższym. (SSMS jest zawarte w Visual Studio, ale wydanie Express wymaga oddzielnego pobierania).

Zależnie od konfiguracji systemu Windows, instalacja i konfiguracja Visual Studio 2013 i SQL Server 2012 mogą wymagać lokalnych uprawnień administratora.

Rozdział 4, „Migrowanie baz danych ” oraz 7, „Microsoft Azure SQL Data Sync”, zawierają procedury, które wymagają lokalnej instancji SQL Server, w której Czytelnik ma uprawnienia do tworzenia baz danych. Jeśli nie dysponuje się takim dostępem, można zainstalować oprogramowanie SQL Server Express Edition (darmową wersję SQL Server), wykonując instrukcje zawarte w kolejnym podrozdziale.

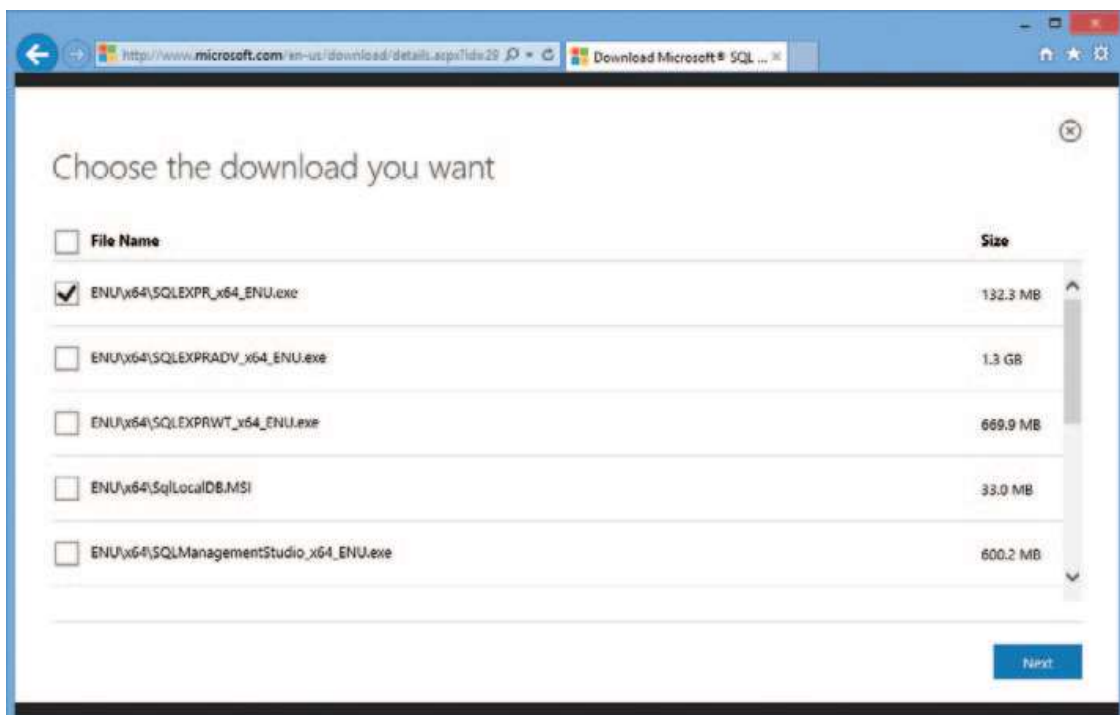
Na koniec poszczególne rozdziały wykorzystują dodatkowe oprogramowanie, które trzeba zainstalować lokalnie. Rozdziały te zawierają szczegółowe procedury pobierania i instalowania tych programów, dzięki czemu możliwe jest wykonanie wszystkich zadań danego rozdziału.

Pobieranie: SQL Server Express Edition

Istnieje wiele wersji SQL Server Express Edition dostępnych do pobrania z witryny firmy Microsoft, przy czym są one udostępnione zarówno jako wersje 32-, jak i 64-bitowe. Można wybrać instalację jedynie silnika bazy danych SQL Server Express (i niczego więcej) albo wybrać jeden z dwóch pozostałych (większych) pakietów: Express With Tools (zawierający SQL Server Management Studio [SSMS]) albo Express With Advanced Services (zawierający SSMS, Full Text Search i Reporting Services). Istnieją również oddzielne pakiety dla SSMS i LocalDB, ale nie zawierają one silnika bazy danych SQL Server Express, niezbędnego do hostowania lokalnych baz danych.

Aby zainstalować silnik bazy danych SQL Server Express Edition, wykonaj poniższe kroki:

1. Otwórz przeglądarkę Internet Explorer i przejdź do adresu <http://www.microsoft.com/en-us/download/details.aspx?id=29062>.
2. Kliknij wielki, pomarańczowy przycisk **Download** (pobierz).
3. Zaznacz odpowiedni pakiet dla swojego systemu, jak na rysunku W-1:
 - a. Dla systemu 64-bitowego wybierz *ENU\x64\SQLEXPRESS_x64_ENU.exe*.
 - b. W przypadku systemu 32-bitowego (lub 64-bitowego, jeśli wolisz uruchamiać aplikację jako WoW) wybierz *ENU\x86\SQLEXPRESS_x86_ENU.exe*.



RYСУNEK W-1 Pobieranie oprogramowania SQL Server 2012 Express Edition (wersja 64-bitowa)



UWAGA Jeśli potrzebujesz pobrać również komponent SQL Server Management Studio (SSMS), wybierz plik *Express With Tools*, czyli taki, który w nazwie zawiera *WT*.

4. Kliknij **Next** (dalej).
5. Jeśli pojawi się monit ostrzeżenia, kliknij **Allow Once** (Zezwalaj raz), jak na rysunku W-2.



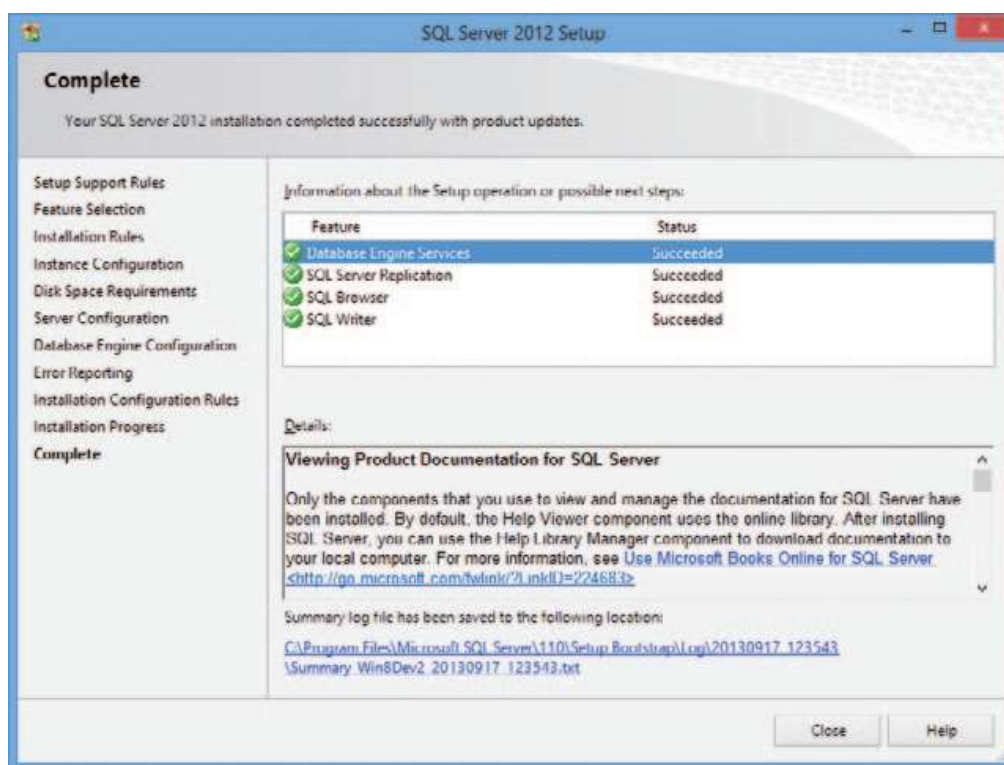
RYSUNEK W-2 Tymczasowe zezwalanie na okno wyskakujące w celu umożliwienia pobierania w razie potrzeby

6. W monicie zapytaniem, czy uruchomić, czy zapisać plik, wybierz **Run** (Uruchom). Spowoduje to rozpoczęcie pobierania.
7. Jeśli pojawi się okno kontroli konta użytkownika po rozpakowaniu pobranych plików, kliknij **Yes** (Tak).
8. W oknie **SQL Server Installation Center** (Centrum instalacji SQL Server) kliknij opcję **New SQL Server Stand-Alone Installation** (Nowa samodzielna instalacja SQL Server), jak na rysunku W-3.



RYSUNEK W-3 Wybieranie nowej instalacji SQL Server

9. W kreatorze instalacji SQL Server 2012 Setup wykonaj poniższe działania:
- Na stronie License Terms zaznacz pole wyboru I Accept The License Terms (akceptując warunki licencji) i kliknij Next.
 - Na stronie Product Updates (aktualizacje produktu) zezwól kreatorowi na wykonanie skanowania w poszukiwaniu aktualizacji, po czym kliknij Next.
 - Na stronie Install Setup Files (instalowanie plików) zaczekaj, aż instalacja się zakończy.
 - Na stronie Feature Selection zaakceptuj domyślne ustawienia, klikając Next.
 - Kontynuuj zatwierdzanie domyślnych ustawień na kolejnych stronach, aż do strony Installation Progress (postęp instalacji) i zaczekaj na zakończenie procesu.
 - Na stronie Complete wskazującej udaną instalację (rysunek W-4) kliknij Close.



RYSUNEK W-4 Zakończona instalacja SQL Server Express

Pobieranie: Przykłady kodu z książki

Wiele rozdziałów zawiera procedury, w których konieczne będzie wpisanie jakiegoś kodu. W większości przypadków są to tylko krótkie fragmenty, ale nadal wygodne będzie pobranie pełnych listingów z witryny powiązanej z książką. W ten sposób łatwiejsze będzie wykonanie procedur, szczególnie tych, w których występuje nieco

więcej kodu do wpisania. Wszystkie fragmenty kodu występujące w książce można pobrać z poniższej strony:

http://aka.ms/AzureSQLDB_SBS

Instalowanie i używanie przykładów kodu

Jeśli na pokazanej wyżej stronie nie jest widoczne łącze Companion Content (powiązana zawartość), należy kliknąć przycisk `other format` (inne formaty) a następnie Companion Content. Pobrany plik zip należy rozpakować. Znajdziemy w nim po jednym folderze dla każdego rozdziału, choć można zauważyć, że nie istnieją żadne przykłady kodu dla rozdziałów 2 i 3. Foldery zawierają pliki .txt odpowiadające numerom listingów w poszczególnych rozdziałach oraz pliki .sql dla procedur składowanych. Dodatkowo foldery dla rozdziałów 6, 8 i 10 zawierają również gotowe (skończone) rozwiązania Visual Studio dla ćwiczeń zawartych w tych rozdziałach.

Podziękowania

Po raz pierwszy poproszono mnie o napisanie książki o SQL Azure – wówczas jeszcze ciągle *nazywało się* to SQL Azure – blisko dwa lata temu. Od tego momentu przebyliśmy długą drogę i pomimo trzęsień ziemi zarówno po stronie platformy produktu Azure, jak i ekosystemu wydawniczego (nie wspominając o innych niespodziankach) szczęśliwie udało nam się dotrzeć do celu!

Jest to moja trzecia książka techniczna i choć każde doświadczenie było unikatowe, z każdej z nich nauczyłem się tego samego: nie można nawet zastanawiać się nad wyzwaniem, jakim jest napisanie książki, bez wsparcia i pomocy licznych, starannych i utalentowanych osób. Są to ci ludzie, którzy zasługują na szczególne wyróżnienie – ci, którzy pomagali mi na tak wiele sposobów, że nie jestem w stanie ustawić ich nazwisk w żadnym sensownym porządku.

Zacznę od Andrew Brusta. Gdyby nie on (a sam jest niekwestionowanym liderem w branży oprogramowania), w ogóle nie zabrałbym się za pisanie książek. Jestem wdzięczny za osobistą przyjaźń jak i za wspólną pracę przy pisaniu i prezentowaniu warsztatów. Te doświadczenia pomogły mi się rozwinąć i urosnąć.

Miałem też szczęście pracować wspólnie z moim kolegą i współautorem, Erikiem Boydem, który napisał cztery rozdziały poświęcone bardzo zaawansowanej tematyce. Eric jest niezwykle utalentowanym programistą, którego doświadczenie i pasja są jawnie widoczne w pisanych przez niego słowach.

Nie można zapomnieć o tych, których wysiłek pozwolił przekształcić surowy tekst w gotową książkę. Russell Jones, mój kumpel z O'Reilly Media, zajmuje specjalną pozycję, gdyż to on namówił mnie do napisania tej książki. Muszę mu podziękować nie tylko za tę okazję, ale również za wskazówki i pomoc podczas przechodzenia

do Microsoft Press. Jeszcze więcej podziękowań otrzymuje Roger LeBlanc za redakcję tekstu i Scott Klein za recenzję techniczną. Nie mogę też pominąć Devona Musgrave i Rosemary Caperton z Microsoft Press oraz Steve'a Sagmana z Waypoint Press. Ich wskazówki były kluczowe dla udanego ukończenia tej książki.

Książka ta nie mogłaby powstać bez wsparcia i miłości mojej rodziny. Dziękuję za to, że byliście tak cierpliwi i tolerancyjni dla mnie, gdy pracowałem nad tym projektem.

Leonard Lobel

Od blisko dwudziestu lat pracuję przy projektowaniu oprogramowania i jestem wdzięczny za to, że miałem możliwość tworzenia, uczenia się i poznawania niekiedy bardzo złożonych, ale fascynujących zagadnień. Pisanie książki takiej jak ta było kolejnym wyzwaniem i nie podołałbym mu bez pomocy i wsparcia wielu ludzi.

Praca nad tym projektem sprawiła mi wiele radości, ale pochłonęła też wiele czasu. Podziękować muszę przede wszystkim mojej żonie Shelly za wszystko, co robi dla naszej rodziny. Chcę też podziękować przyjaciołom i współpracownikom za ich wsparcie podczas tej pracy.

Oczywiście, ważne miejsce zajmuje Lenni Lobel, który zaprosił mnie do dołączenia do tego projektu. Lenni jest fantastycznym współautorem i wykonał świetną robotę, prowadząc ten projekt aż do ukończenia. Jego wskazówki i pomoc w znajdowaniu właściwego formułowania myśli były niezwykle cenne zarówno dla mnie samego, jak i dla całego projektu.

Na koniec, ale na pewno nie najmniej ważne, są wszystkie osoby w firmie Microsoft i Microsoft Press, które pomagały w realizacji tego projektu. Lista ta (niewątpliwie niekompletna) obejmuje Scotta Kleina, Dorę Chan, Marka Browna, Devona Musgrave, Rosemary Caperton, Steve'a Sagman, Conora Cunninghama i cały zespół Azure CAT.

Eric Boyd

Errata, poprawki i wsparcie dla książki

Dołożyliśmy wszelkich starań aby zagwarantować poprawność tej książki i powiązanej z nią zawartości. Aktualizacje do tej książki w formie erraty zawierającej zgłoszone usterki i ich poprawki są dostępne na karcie Errata & Updates strony internetowej dotyczącej książki pod adresem:

http://aka.ms/AzureSQLDB_SBS

W razie zauważenia błędu, który nie jest jeszcze wymieniony, można go zgłosić za pośrednictwem tej samej strony.

Dodatkową pomoc można uzyskać z działu Microsoft Press Book Support pod adresem mspinput@microsoft.com.

Prosimy zauważyć, że wsparcie dotyczące oprogramowania i sprzętu firmy Microsoft nie jest oferowane pod powyższymi adresami. Pomoc dotyczącą produktów firmy Microsoft można uzyskać pod adresem <http://support.microsoft.com>.

Czekamy na uwagi

Najwyższym priorytetem wydawnictwa Microsoft Press jest zadowolenie Czytelników, a ich opinie są dla nas niezwykle cenne. Prosimy o przekazywanie uwag w witrynie:

<http://aka.ms/tellpress>

Ankieta jest krótka, a my czytamy wszystkie przysłane do nas spostrzeżenia i pomysły. Z góry dziękujemy za wszystkie wpisy!

Pozostańmy w kontakcie

Niech dyskusja trwa! Jesteśmy na Twitterze: <http://twitter.com/MicrosoftPress>.

ROZDZIAŁ 1

Poznajemy Microsoft Azure SQL Database

Leonard Lobel

W tym rozdziale wspólnie z Czytelnikiem utworzymy pierwszą bazę danych na platformie Microsoft Azure SQL Database – całkowicie *od zera*. Należy przez to rozumieć, że wszystko, co będzie potrzebne, aby wykonać opisane tu zadania, to przeglądarka sieci Web (w tym rozdziale użyty został Internet Explorer) oraz dostęp do Internetu. Czytelnik uzyska konto Microsoft (jeśli go jeszcze nie ma) i posłuży się nim w celu utworzenia i uzyskania dostępu do darmowej próbnej subskrypcji usługi Microsoft Azure. Następnie poznamy portal zarządzający Microsoft Azure, a następnie zapewnimy, aby serwer i baza danych rozpoczęły działanie w chmurze. Na koniec posłużymy się portalem zarządzania SQL Database do zaprojektowania, wypełnienia i odpytowania bazy danych.

UWAGA W tej książce będziemy się zazwyczaj odwoływać do Microsoft Azure SQL Database jako do *SQL Database*. Termin *SQL Server* będzie zawsze oznaczał instancje bazy danych zainstalowane w siedzibie firmy (*on-premises*), podczas gdy *SQL Database* oznacza zawsze bazę danych opartą na chmurze Microsoft Azure.



Choć książka ta koncentruje się głównie na Microsoft Azure SQL Database, powinna być pomocna w poznawaniu szerszego kontekstu platformy Microsoft Azure i w ogólności przetwarzania w chmurze. Wiedza taka pozwoli lepiej zrozumieć wartość SQL Database. Tak więc, zanim zajmiemy się tworzeniem konta Microsoft, pozwolę sobie na skrótowe omówienie przetwarzania w chmurze ze szczególnym uwzględnieniem usługi Microsoft Azure.

Przetwarzanie w chmurze: przedstawienie pojęcia

Microsoft Azure to należąca do firmy Microsoft platforma przetwarzania w chmurze (*cloud-computing*). Jednak czym właściwie jest *przetwarzanie w chmurze*? Problem w tym, że nie istnieje precyzyjna definicja. W rzeczywistości termin ten jest niejednoznaczny. Istnieje dziś wiele różnych typów usług, które działają „w chmurze”. Zasadniczo termin *przetwarzanie w chmurze* odnosi się do ewolucji, jaką przeszły internetowe usługi hostingowe, prowadząc do sytuacji, w której dostawcy (tacy jak Microsoft, Google, Amazon i wielu innych) oferują indywidualnym konsumentom i firmom usługi realizowane na sprzęcie zapewniającym niezbędną nadmiarowość, przy czym obsługa tych systemów jest częściowo lub w pełni zautomatyzowana (z punktu widzenia klienta). Ten poziom usług znacznie wykracza poza tradycyjny hosting internetowy, który pojawił się wraz z „bąblem dot-comów” w latach dziewięćdziesiątych XX wieku. W przypadku przetwarzania w chmurze Internet nie jest już jedynie używany jako medium dla udostępniania informacji. W istocie chmura wykorzystuje Internet jako sposób łączenia klientów z różnymi usługami infrastruktury, platformy lub aplikacji na znacznie większym poziomie elastyczności i abstrakcji, niż możliwy do zaoferowania przez wcześniej stosowane schematy hostingu.

Jedną z najwcześniejszych platform przetwarzania w chmurze były Amazon Web Services (AWS), wprowadzone w 2002 roku przez Amazon.com. Także dziś AWS jest jednym z głównych graczy w branży przetwarzania w chmurze. Począwszy od połowy pierwszej dekady XXI wieku przetwarzanie w chmurze zaczęło szybko zdobywać popularność i w roku 2009 firma Microsoft zaprezentowała swoje rozwiązanie Windows Azure, które w kwietniu 2014 zostało przemianowane na Microsoft Azure. Od samego powstania Azure firma Microsoft stale rozszerza swoją platformę o nowsze i bogatsze możliwości.

Natychmiastowe, dynamiczne wdrażanie

Zacznijmy od tego, że wdrażanie i konfigurowanie własnych (zainstalowanych w siedzibie firmy) serwerów jest trudne. Po pierwsze, musimy zakupić i fizycznie rozmieścić odpowiedni sprzęt. Następnie trzeba uzyskać niezbędne licencje na oprogramowanie, zainstalować system operacyjny, po czym wdrożyć i skonfigurować potrzebne oprogramowanie. Konieczne jest też zapewnienie akceptowalnego poziomu wydajności i ciągłość działania na wypadek nieoczekiwanej awarii sprzętu, oprogramowania i sieci. Oznacza to konfigurowanie równoważenia obciążeń i nadmiarowości poprzez technologie duplikowania i klastrowania. Trzeba też zdefiniować strategię tworzenia kopii zapasowych i stosować ją ściśle jako część ogólnego planu przywracania po awarii – który również trzeba będzie opracować. A gdy już wszystko zostanie skonfigurowane

i uruchomione, nadal trzeba będzie o to dbać, aby zapewnić sprawne działanie systemu przez cały czas życia naszej aplikacji.

Bez zbędnej przesady, *przejście do chmury eliminuje wszystkie te obciążenia*.

W skrócie, idea aplikacji i usług uruchamianych w „chmurze” oznacza, że mamy do czynienia z „niedotykalnymi” zasobami sprzętowymi, co z kolei przekłada się na środowisko robocze pozbawione konieczności jego konserwowania. Rejestrujemy się w firmie hostującej chmurę (Microsoft w przypadku Azure), płacimy jej za tyle mocy (w znaczeniu zasobów), ile potrzebuje nasza aplikacja (RAM, CPU, pojemność magazynu, pasmo przenoszenia, skalowalne równoważenie obciążenia i tak dalej) i to ta firma martwi się o całą resztę. W porównaniu z ilością pracy i potencjalnymi błędami, które mogą pojawić się przy samodzielnym robieniu tego wszystkiego, rozwiązanie takie jest wolne zarówno od mordęgi, jak i ryzyka.

Chmura Microsoft Azure

W przypadku Azure nasze aplikacje, usługi i dane rezydują w chmurze. Chmura Azure oparta jest na wielkich, geograficznie rozproszonych centrach danych należących do firmy Microsoft, wyposażonych w wydajne serwery, ogromne pojemności magazynu danych i bardzo wysoki poziom nadmiarowości, zapewniający ciągłość działania.

Jednak środowisko Microsoft Azure to znacznie więcej, niż zbiór konwencjonalnych serwerów hostingowych „na sterydach”. W istocie nasze chmurowe aplikacje i usługi nie są uruchamiane bezpośrednio na tych serwerach. Zamiast tego na tym całym sprzęcie działa zaawansowana technologia wirtualizacji – *hyperwizor*. Nasz „kod w chmurze” z kolei działa na tej warstwie wirtualizacji. Dzięki temu przeskalowanie w czasie szczytowego obciążenia sprowadza się do zmiany ustawień konfiguracyjnych, które zwiększają liczbę uruchomionych instancji w celu obsługi rosnących potrzeb. Gdy gorący sezon się skończy, analogiczna prosta zmiana pozwala zredukować liczbę instancji i przeskalowanie rozwiązania w dół. Microsoft Azure obsługuje skalowalność poprzez dynamiczne przyznawanie większej lub mniejszej mocy obliczeniowej zwirtualizowanemu środowisku. Proces ten, często nazywany *elastycznym skalowaniem*, jest praktycznie natychmiastowy.

A teraz spróbujmy rozważyć ten sam scenariusz przy konwencjonalnej infrastrukturze. Musimy kupić i zainstalować dodatkowe serwery, włączyć je i dodać jako członków farmy równoważenia obciążenia. A gdy dodatkowa moc nie będzie już potrzebna, trzeba je odłączyć do trybu offline, aby czekały na kolejny szczyt obciążenia. Wymaga to wiele pracy i czasu – naszej własnej lub firmy hostingowej – zbyt wiele, gdy porównamy to z dostosowywaniem konfiguracji poprzez kilka kliknięć myszą.

Ponieważ rozwiązania chmurowe mogą być dostarczane na wiele różnych sposobów, w słowniku pojawiło się wiele nowych słów i sloganów. Wyróżniają się wśród nich najrozmaitsze akronimy pochodne od zwrotu „as a service” (jako usługa), takie jak IaaS (Infrastructure as a Service – infrastruktura jako usługa), PaaS (Platform

as a Service – platforma jako usługa) oraz SaaS (Software as a Service – oprogramowanie jako usługa). Wszystkie te terminy jednoznacznie odwołują się do *usług (service)*; różnica leży w *poziomiej* tej usługi. Zazwyczaj pomocne jest myślenie o tych terminach jako o różnych stopniach abstrakcji, poczynając od najniższego poziomu infrastruktury sprzętowej. W przypadku instalacji w siedzibie nie ma żadnej abstrakcji i jesteśmy odpowiedzialni za wszystkie poziomy, od sprzętu w górę. Po przejściu do chmury możemy wybrać wariant IaaS, Paas lub SaaS, zgodnie z potrzebami, przy czym każde z tych podejść zapewnia większy poziom abstrakcji.

Infrastruktura jako usługa

W wariantcie IaaS Microsoft Azure w praktyce oznacza udostępnienie maszyn wirtualnych (VM), które są w całości pod naszą kontrolą. Podobnie jak w środowisku tradycyjnym, jesteśmy odpowiedzialni za instalację systemu operacyjnego i skonfigurowanie maszyny. Łatwo można zbudować maszynę wirtualną od zera – albo dostosować istniejące maszyny wirtualne z biblioteki wstępnie skonfigurowanych VM – po czym je wdrożyć do działania w chmurze z pełnymi możliwościami sieciowymi (włącznie z połączeniami Virtual Private Network [VPN]). Jednak w odróżnieniu od pracy z serwerami w siedzibie, nigdy nie będą nam potrzebne śrubokręty, dyski, kable, szafy rackowe, zasilacze, płyty główne, pamięć ani nic podobnego. Na tym właśnie polega IaaS – jest abstrakcją sprzętu (sieć, magazyn, serwery, wirtualizacja) i niczym więcej.

Mając takie możliwości, możemy oczywiście utworzyć VM w Microsoft Azure, na której będzie działać Microsoft SQL Server; właśnie tak, maszynę wirtualną w chmurze, na której uruchomimy pełną, „*on-premises*” wersję SQL Server. Mogą istnieć sytuacje, gdy takie działanie będzie w pełni usprawiedliwione i poprawne – na przykład gdy wymagana będzie pełna kompatybilność z istniejącą już w siedzibie firmy instalacją SQL Server, czego nie zapewnia SQL Database. (Różnice pomiędzy tymi dwiema platformami omówimy szczegółowo w rozdziale 3, „Różnice pomiędzy SQL Server a Microsoft Azure SQL Database”). Typowym przykładem takiego scenariusza jest potrzeba dostarczenia usługi raportowania z SQL Server Reporting Services (SSRS) uruchomioną na maszynie wirtualnej Microsoft Azure (jak to zrobić, nauczymy się w rozdziale 6, „Raportowanie w chmurze”). Trzeba jednak pamiętać, że uruchomienie SQL Server na maszynie wirtualnej Azure jest czymś zupełnie odmiennym od korzystania z Microsoft Azure SQL Database. Wybranie IaaS i SQL Server oznacza, że nadal jesteśmy w pełni odpowiedzialni za obsługę naszej maszyny wirtualnej (lub maszyn) w chmurze. Obejmuje to instalację systemu operacyjnego od zera (chyba że wybierzemy załadowanie wstępnie skonfigurowanej VM z galerii), instalację SQL Server, skonfigurowanie instancji, zapewnienie aktualności oprogramowania, ochronę maszyn wirtualnych przed awariami programowymi i tworzenie kopii zapasowych. Jeśli nasza maszyna wirtualna ulegnie awarii, wówczas to *my* mamy problem. SQL Database jest czymś zupełnie odmiennym, i to właśnie czyni ją rozwiązaniem typu PaaS.

Platforma jako usługa

W przypadku PaaS poziom abstrakcji został podniesiony znacznie wyżej w porównaniu do IaaS, przez co jesteśmy odgraniczeni od warstw systemu operacyjnego, warstwy pośredniej (*middleware*) i czasu wykonania (*runtime*). Oznacza to, że Microsoft Azure zapewnia *platformę* dla naszych aplikacji i usług, na której możemy je uruchomić. Nie mamy żadnej kontroli nad samą platformą; musimy zarządzać tylko aplikacjami i danymi, podczas gdy dostawca chmury zarządza resztą infrastruktury. Nadal musimy samodzielnie utworzyć i przetestować nasze aplikacje, po czym załadować je do Microsoft Azure. (Zagadnienie to omówimy w rozdziale 10, „Budowanie rozwiązania w chmurze”). Zapewnia to naszej aplikacji niewiarygodną skalowalność bez konieczności inwestowania w kosztowny sprzęt, co byłoby wymagane w dowolnym scenariuszu instalacji w siedzibie.

SQL Database jest rozwiązaniem typu PaaS. To nadal jest SQL Server, ale w celu dostarczenia *platformy* dla relacyjnej bazy danych (w przeciwieństwie do infrastruktury) określone funkcjonalności, które są dostępne w rozwiązaniach w siedzibie, nie są obsługiwane. W przypadku SQL Database można skonfigurować serwery i bazy danych „w locie”, bez żadnej interakcji z systemem operacyjnym czy innymi elementami podrzędnej infrastruktury. Nie musimy nawet wiedzieć ani zastanawiać się nad tym, czy dane i pliki dzienników są przechowywane na dysku C, czy D, gdyż SQL Database obsługuje za nas wszystkie szczegóły fizycznego przechowywania danych. I jak zobaczymy w rozdziale 3, korzyści z praktycznie natychmiastowego wdrażania oraz bezwysiłkowa i pozbawiona ryzyka obsługa oznacza też pewną utratę kontroli, którą mamy przy wykorzystywaniu instalacji SQL Server w siedzibie.

Oprogramowanie jako usługa

SaaS (Software as a Service) to najwyższy poziom abstrakcji, w którym wszystko – od poziomu sprzętu, aż po aplikacje końcowych użytkowników – jest obsługiwane przez usługę. Obecnie istnieje już wiele chmur typu SaaS, w tym Office 365, CRM oraz Salesforce.com.

Możemy utworzyć nasze własne rozwiązanie SaaS oparte na SQL Database, wspierając usługę lub witrynę Web – również hostowaną w Azure – na tej bazie danych. (Zrobimy to w rozdziale 10). Następnie będziemy mogli zaoferować kompletne rozwiązanie naszym klientom, którzy będą wchodzić w interakcję z aplikacją korzystając jedynie z przeglądarki lub urządzenia mobilnego. Klienci ci nie muszą (i nie potrzebują) znać żadnych aspektów infrastruktury ani platformy – po prostu łączą się z aplikacją. Tak więc, z ich perspektywy, dostarczamy im prawdziwe rozwiązanie SaaS.

Rejestrowanie się dla SQL Database

Aby móc zacząć korzystać z SQL Database, potrzebujemy dwóch rzeczy: konta Microsoft oraz subskrypcji Microsoft Azure. Jest całkiem możliwe, że Czytelnik już ma konto Microsoft, znane wcześniej jako Windows Live ID. Jest to to samo konto, którego używamy dziś do logowania się w różnych witrynach i usługach firmy Microsoft, takich jak Outlook.com, Hotmail, Xbox LIVE, Windows Phone, OneDrive (dawniej SkyDrive) i wielu innych.

Tworzenie konta Microsoft

Pierwszym krokiem, który trzeba wykonać, zanim będzie można użyć dowolnej z usług Microsoft Azure, jest uzyskanie konta Microsoft, o ile jeszcze go nie mamy. Zasadniczo jest to kombinacja adresu email i hasła, którego użyjemy do utworzenia subskrypcji Microsoft Azure, a później do uzyskiwania dostępu do Azure. Jeśli Czytelnik ma już konto Microsoft, może użyć go do utworzenia nowej subskrypcji Azure. Nie ma potrzeby tworzenia konta, zatem w takiej sytuacji można przejść od razu do następnego podrozdziału, „Tworzenie subskrypcji Microsoft Azure”. W przeciwnym razie konieczne będzie utworzenie konta.



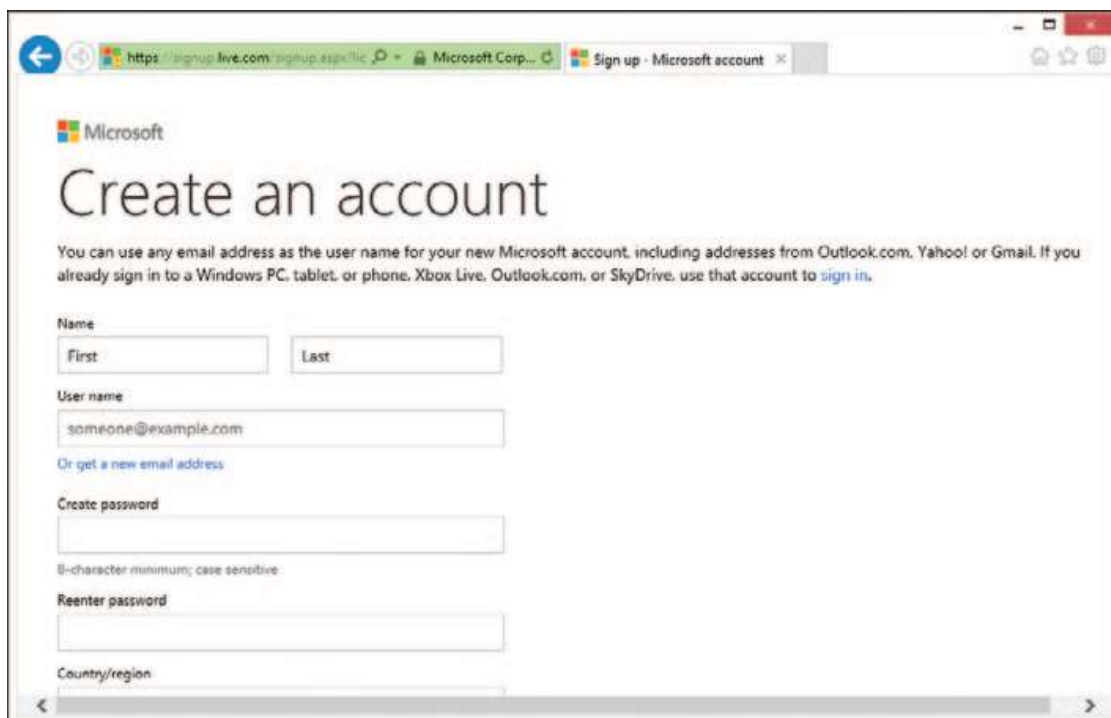
Wskazówka Jeśli Czytelnik ma już konto Microsoft, ale chce z dowolnego powodu użyć innego adresu email, nadal nie ma potrzeby tworzenia nowego konta. Można bowiem albo przemianować istniejące konto, albo utworzyć alias. Więcej informacji zawiera strona <http://windows.microsoft.com/en-US/hotmail/get-new-outlook-address>.

Przy tworzeniu nowego konta Microsoft nazwą użytkownika może być adres email, który już mamy. Alternatywnie można utworzyć nowy adres email dla nowego konta, który będzie się kończył albo na *@outlook.com*, albo *@hotmail.com*. W rzeczywistości nie ma znaczenia, co wybierzemy, o ile wybrana nazwa nie jest już zajęta przez kogoś innego w jednej z domen *@outlook.com* lub *@hotmail.com*. Jeśli zdecydujemy się na utworzenie nowego adresu, dostaniemy również skrzynkę pocztową pod tym adresem i firma Microsoft będzie się z nami komunikowała poprzez ten adres za każdym razem, gdy pojawi się potrzeba przekazania jakiejś ważnej informacji dotyczącej konta.

Bez względu na to, czy wybierzemy istniejący adres email, czy utworzymy nowy, musimy również przypisać do tworzonego konta silne hasło. Wymagane są również pewne dodatkowe informacje osobiste, takie jak nazwisko i imię, płeć, jedną z dwóch form potwierdzenia tożsamości, kraj oraz kod pocztowy.

Aby utworzyć nowe konto Microsoft, wykonaj poniższe czynności:

1. Korzystając z przeglądarki Internet Explorer przejdź do adresu <http://signup.live.com>. Pojawi się strona **Create An Account** (utwórz konto), pokazana na rysunku 1-1.



RYСУNEK 1-1 Rejestrowanie nowego konta Microsoft

2. Wpisz swoje imię i nazwisko.
3. Jako nazwę użytkownika konta Microsoft (tego konta będziemy używać do logowania się w portalu Microsoft Azure) podaj istniejący adres email albo kliknij łącze **Or get a new email address** (albo uzyskaj nowy adres email), by utworzyć nowy adres w domenie @outlook.com lub @hotmail.com.
4. Wpisz hasło, a następnie wpisz je ponownie w celu potwierdzenia. Hasło dla konta musi być silne, o długości co najmniej ośmiu znaków i zawierać kombinację małych i wielkich liter, cyfr i symboli.
5. Wybierz z odpowiednich list nazwę kraju i kod pocztowy, datę urodzenia oraz płeć.
6. Podaj numer telefonu lub alternatywny adres email. Konieczne jest wskazanie przynajmniej jednej spośród tych metod potwierdzania tożsamości.
7. Wpisz pokazane niżej przypadkowe znaki (CAPTCHA) w celu sprawdzenia, że jesteś rzeczywistą osobą, a nie botem.
8. Kliknij przycisk **Create Account** (utwórz konto).

Jeśli w kroku 3 został utworzony nowy adres email, nastąpi natychmiastowe przekierowanie do strony **Account Summary** (podsumowanie konta). W przypadku podania istniejącego konta po chwili zostanie nań wysłana wiadomość od nadawcy „Microsoft account team”. Wiadomość ta ma na celu weryfikację, że w istocie jesteśmy właścicielami podanego adresu pocztowego. Nowe konto Microsoft nie zostanie uaktywnione, dopóki nie klikniemy weryfikacyjnego łącza zawartego w tej wiadomości.

Tworzenie subskrypcji Microsoft Azure

Teraz, gdy już mamy konto Microsoft, możemy przejść do tworzenia subskrypcji Azure. Subskrypcja ta jest zasadniczo kontem rozliczeniowym Microsoft Azure i otwiera drzwi do pełnego zakresu usług dostępnych w Microsoft Azure – w tym oczywiście SQL Database.

W pokazanej dalej procedurze utworzymy darmową próbną subskrypcję w Microsoft Azure. W momencie pisania tych słów darmowa próbna subskrypcja obejmowała kredyt w wysokości 200 dolarów (150 euro) na 30-dniowy dostęp do wszystkich usług. Proces ten wymaga podania danych karty kredytowej, która zostanie użyta do opłacenia subskrypcji po wygaśnięciu okresu próbnego¹.

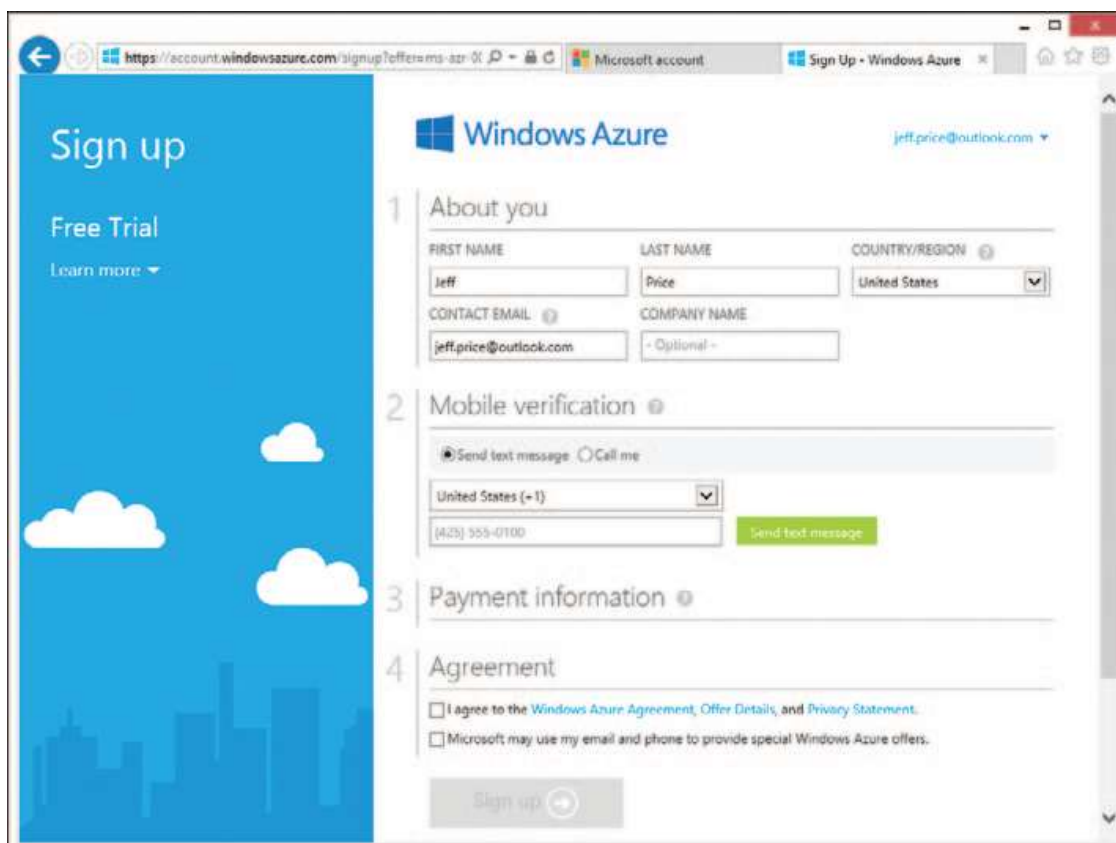


WAŻNE Ceny usług Microsoft Azure i oferty specjalne podlegają ciągłym zmianom. Zdecydowanie polecamy odwiedzenie strony <http://www.windowsazure.com/pl-pl/pricing/purchase-options/> w celu sprawdzenia aktualnych cen. Dodatkowo subskrybenci MSDN mają do dyspozycji specjalne oferty. Więcej informacji można znaleźć pod adresem <http://www.windowsazure.com/pl-pl/pricing/member-offers/msdn-benefits/>.

Wykonaj poniższe czynności, aby utworzyć nową subskrypcję Microsoft Azure:

1. Przy użyciu przeglądarki Internet Explorer przejdź do adresu <http://www.windowsazure.com>.
2. Kliknij zielony przycisk **Try For Free** (Wypróbuj bezpłatnie).
3. Na kolejnej stronie kliknij zielony przycisk **Try It Now** (Wypróbuj teraz).
4. Jeśli nie byłeś jeszcze zalogowany na konto Microsoft, zaloguj się teraz.
5. Zostaniesz przekierowany do strony **Sign up** (Utwórz konto), pokazanej na rysunku 1-2.

¹ Witryna rejestracji Microsoft Azure jest dostępna również w języku polskim i w przypadku komputera zlokalizowanego na terenie Polski może nastąpić automatyczne przekierowanie do tej wersji. Zasadniczy portal Azure i inne strony pochodne są dostępne w kilku językach (do wyboru podczas rejestracji), ale nie ma wśród nich wersji polskiej. (Wszystkie przypisy pochodzą od tłumacza).



RYSUNEK 1-2 Rejestrowanie nowej bezpłatnej wersji próbnej subskrypcji Microsoft Azure

6. Wybierz metodę odebrania kodu weryfikacyjnego – jako wiadomość SMS lub rozmowę telefoniczną.
7. Wpisz kod otrzymany w wiadomości tekstowej lub w rozmowie i kliknij **Verify Code** (Zweryfikuj kod).
8. Podaj dane karty kredytowej, która zostanie użyta do realizacji płatności po wygaśnięciu okresu próbnego.

WAŻNE Podanie danych karty kredytowej nie oznacza, że zostaniemy obciążeni jakąkolwiek opłatą, o ile nie odblokujemy subskrypcji. Dane te mają na celu przede wszystkim weryfikację tożsamości użytkownika.



9. Zaznacz pole wyboru, aby potwierdzić zgodę na warunki umowy.
10. Kliknij przycisk **Sign Up** (Utwórz konto).

Konfigurowanie nowej subskrypcji Azure zajmuje tylko kilka sekund i już po chwili możemy zacząć pracę z SQL Database i dowolnymi innymi usługami Microsoft Azure.

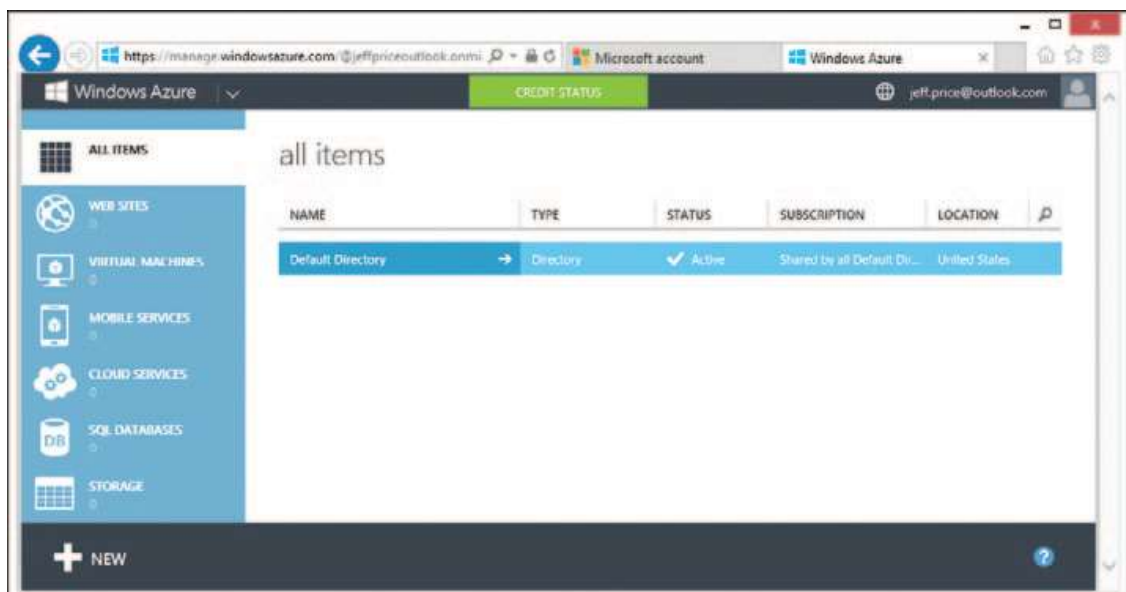
Tworzenie serwera

Tworzenie *serwera* jest proste. Serwer jest podobny do instancji SQL Server w tym sensie, że może hostować wiele baz danych. Wszystko, co trzeba zrobić to utworzenie konta administratora z silnym hasłem i wskazanie regionu geograficznego, w którym serwer powinien zostać fizycznie zlokalizowany. W celu osiągnięcia najlepszej wydajności należy wybrać region możliwie bliski naszym klientom. Jak pokażemy w rozdziale 2, „Konfiguracja i ceny”, będziemy również chcieli się upewnić, że dowolne witryny i usługi chmury Microsoft Azure (takie jak te, które utworzymy w rozdziale 10) będą hostowane w tym samym regionie, co serwery SQL Database, z którymi będą się łączyć. Dzięki umieszczeniu obydwu części rozwiązania w tym samym regionie unikniemy opłat opartych na paśmie przesyłowym, które pojawiają się, gdy poszczególne elementy muszą się komunikować poprzez granice regionów Azure. Co również ważne, w ten sposób redukujemy opóźnienia, co zapewnia znacząco lepszą wydajność.

SQL Database używa również specjalnych *reguł zapory*, które można ustawić w celu kontrolowania, który komputer lub komputery mogą uzyskiwać dostęp do naszego serwera bazy danych w chmurze. Jako minimum trzeba dodać regułę przyznającą dostęp z adresu IP własnego komputera, aby móc uzyskiwać dostęp do serwera z komputera lokalnego. W rozwiązaniu produkcyjnym konieczne może być dodanie reguł przyznających dostęp dla jednego lub kilku bloków adresów IP. Więcej informacji na temat reguł zapory przedstawimy w rozdziałach 2 i 5.

Aby utworzyć nowy serwer, wykonaj poniższe kroki:

1. Zaloguj się do portalu Microsoft Azure pod adresem <https://manage.windowsazure.com>. Zostaniesz przekierowany do głównej strony portalu prezentującej all items (wszystkie elementy), pokazanej na rysunku 1-3.

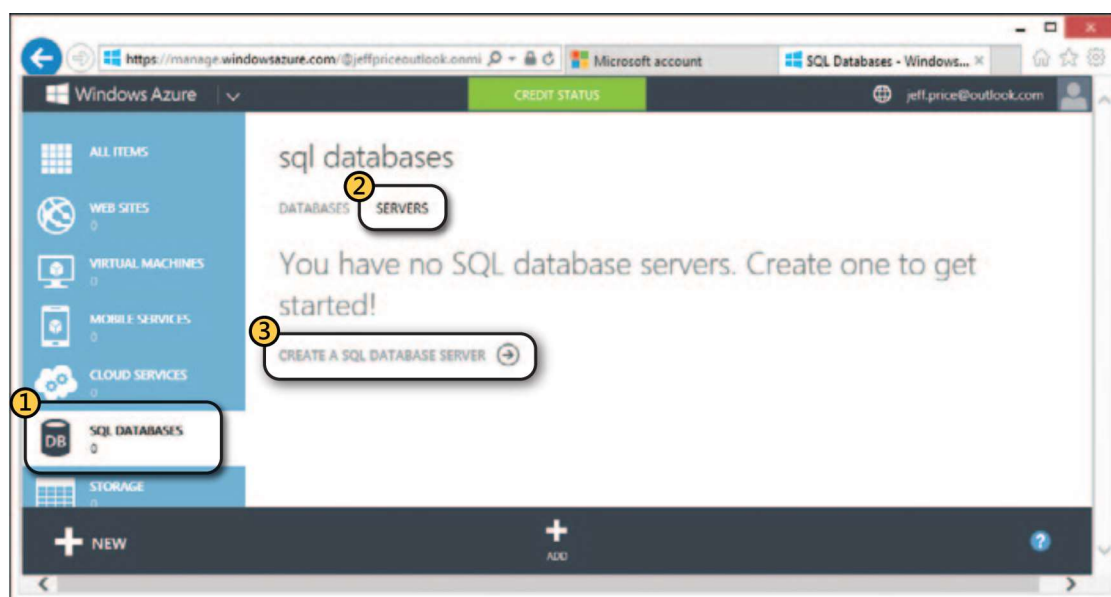


RYSUNEK 1-3 Portal zarządzania Microsoft Azure bez żadnych skonfigurowanych usług

UWAGA Przy pierwszym logowaniu w portalu pojawi się zaproszenie do przejrzania oferowanych usług. Możesz przyjąć to zaproszenie albo zamknąć komunikat, by przejść do głównej strony portalu.

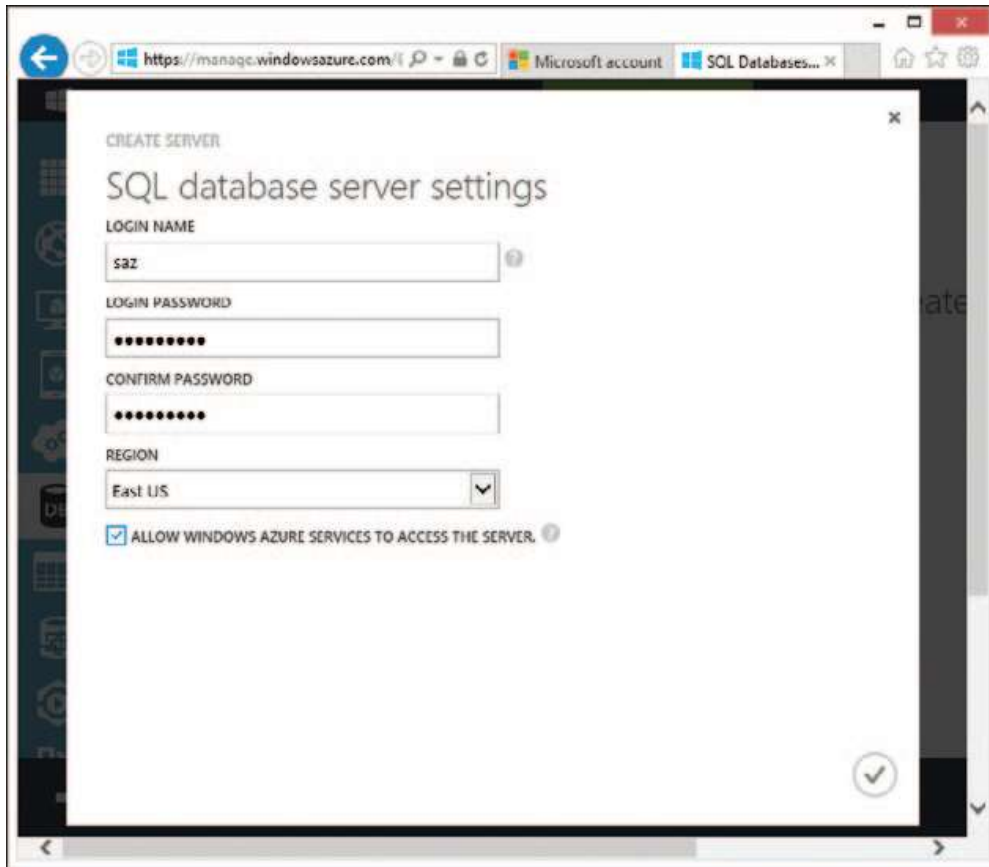


2. Kliknij **SQL DATABASES** w pionowym panelu nawigacyjnym po lewej (rysunek 1-4), później **SERVERS** w górnej części głównego panelu i następnie **CREATE A SQL DATABASE SERVER** (utwórz serwer SQL Database).



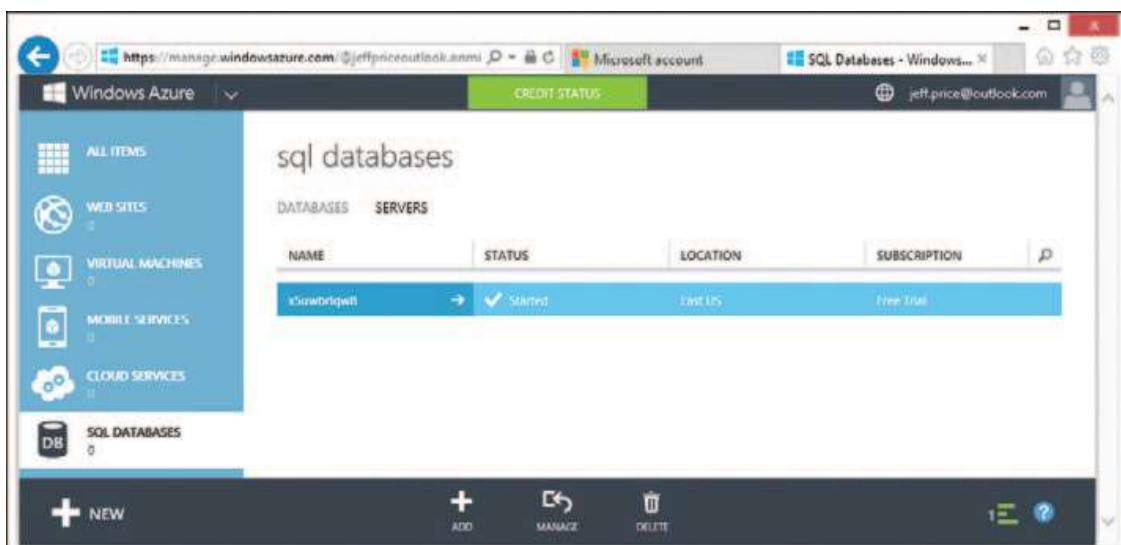
RYСУNEK 1-4 łącze CREATE A SQL DATABASE SERVER na stronie SQL DATABASES

3. Wpisz nazwę logowania dla nowego serwera – na przykład *saz*.
4. Wpisz hasło dla nowego serwera i powtórz je w drugim polu dla potwierdzenia. Hasło musi stosować się do typowych reguł silnego hasła, czyli zawierać kombinację małych i wielkich liter, cyfr i symboli.
5. Wybierz odpowiedni region z listy rozwijanej – na przykład **West Europe** (zachodnia Europa). Aby zapewnić najlepszą wydajność, należy wybrać region, w którym się znajdujesz lub możliwie najbliższy.
6. Upewnij się, że pole wyboru **ALLOW WINDOWS AZURE SERVICES TO ACCESS THE SERVER** (zezwalaj usługom Windows Azure na dostęp do serwera) jest zaznaczone. Spowoduje to, że serwer będzie dostępny dla usług chmury Microsoft Azure, które będziemy tworzyć lub z nich korzystać w innych rozdziałach (przypomnijmy, że Microsoft Azure było wcześniej nazywane Windows Azure). Strona powinna wyglądać podobnie do pokazanej na rysunku 1-5.



RYSUNEK 1-5 Okno dialogowe CREATE SERVER

7. Kliknij ikonę zatwierdzenia (z „ptaszkiem”) w prawym dolnym rogu okna dialogowego, aby zatwierdzić ustawienia. Po kilku chwilach nowy serwer zostanie skonfigurowany i udostępniony do użycia, jak na rysunku 1-6.



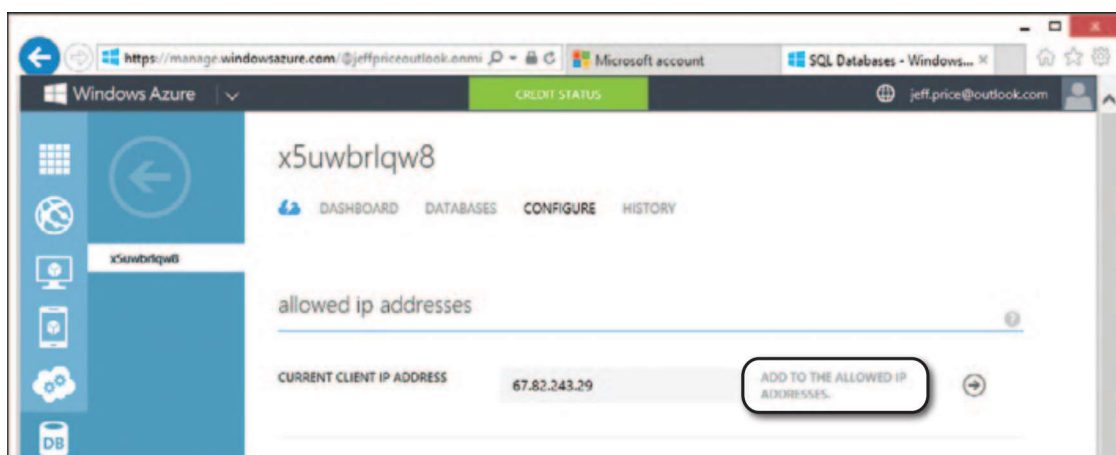
RYSUNEK 1-6 Nowy serwer SQL Database

Ktoś, kto choć raz samodzielnie przygotowywał nowy serwer w siedzibie firmy, zdecydowanie doceni zaoszczędzony właśnie czas i wysiłek. Serwer jest już dostępny i gotowy do przechowywania bazy danych, zaś mechanizm SQL Database automatycznie przypisał mu losową i unikatową (choć względnie krótką) nazwę, po której można uzyskiwać do niego dostęp. Jednak zanim dostęp zostanie przyznany, musi zostać skonfigurowana zapora serwera. Zatem następnym krokiem będzie dodanie reguł zapory, abyśmy mogli połączyć się z serwerem z naszego komputera lokalnego.

Pole wyboru wspomniane w punkcie 6 poprzedniej procedury powoduje dodanie do reguł zapory specjalnego adresu IP 0.0.0.0, który pozwala na dostęp do naszego serwera usługom chmury działającym w Microsoft Azure. Nadal jednak musimy dołączyć adres IP naszego lokalnego komputera, aby móc uzyskać dostęp do serwera z portalu zarządzania SQL Database i innych narzędzi (takich jak SQL Server Management Studio lub SQL Server Data Tools, zawarte w Microsoft Visual Studio, o których dowiemy się więcej w późniejszych rozdziałach).

Aby dodać regułę zapory dla adresu IP maszyny lokalnej, wykonaj następujące działania:

1. Kliknij nazwę serwera, a następnie łącze CONFIGURE (konfiguruj) u góry strony.
2. Kliknij łącze ADD TO THE ALLOWED IP ADDRESSES (dodaj do dozwolonych adresów IP) obok wykrytego bieżącego adresu komputera, jak na rysunku 1-7. Zostanie utworzona nowa reguła zapory dla adresu IP twojego komputera.



RYСУNEK 1-7 Dodawanie adresu IP maszyny lokalnej do listy dozwolonych adresów w regułach zapory

3. Kliknij SAVE (zapisz) w dolnej części strony.
4. Kliknij ikonę powrotu (wielką strzałkę skierowaną w lewo), aby powrócić do strony SQL DATABASES dla nowego serwera.

Może być konieczne odczekanie kilku chwil, aby nowa reguła zapory zaczęła działać, ale zazwyczaj odbywa się to bardzo szybko (w granicach od pięciu do dziesięciu sekund). Jeśli jednak nie zaczekamy dostatecznie długo i reguła jeszcze nie zaczęła być

używana, możemy być pewni, że nie uda się nam połączyć z serwerem z komputera lokalnego, dopóki nowa reguła nie zadziała.

Tworzenie instancji SQL Database

Tworzenie bazy danych jest niemal tak proste, jak tworzenie serwera. Trzeba po prostu wybrać nazwę dla nowej bazy, jej wydanie, maksymalną wielkość, domyślne ustawienia językowe dla sortowania (*collation*) i, oczywiście, serwer, na którym baza ma być hostowana.

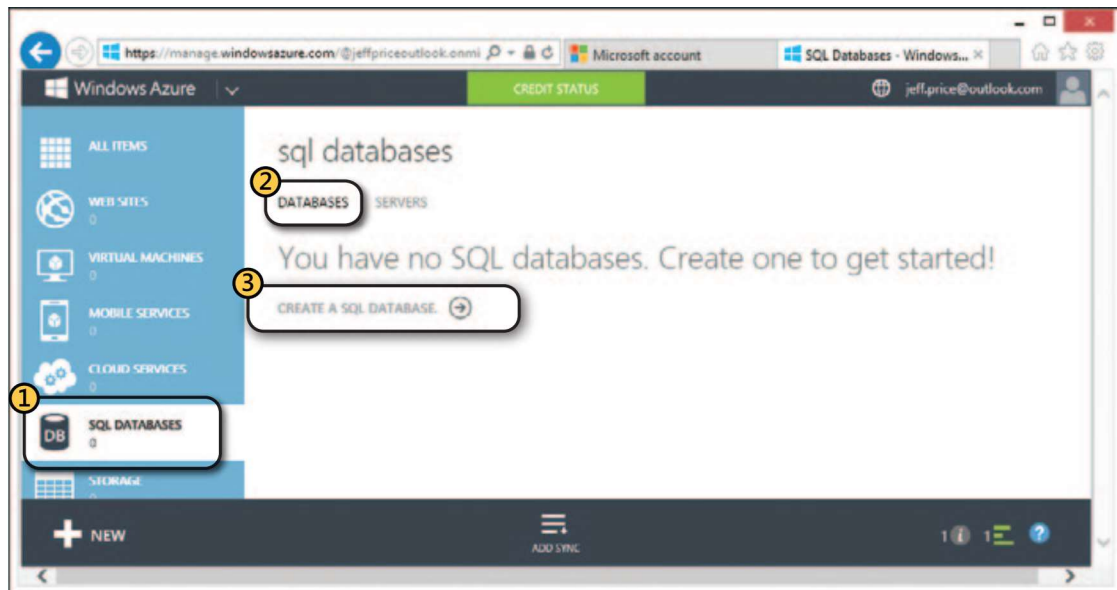
W rozdziale 2 poznamy więcej różnych opcji dotyczących wydania bazy danych i maksymalnego rozmiaru. Na razie ważne jest, aby wiedzieć, że wszystkie te ustawienia (z wyjątkiem sortowania domyślnego) można łatwo zmienić w późniejszym terminie. W ramach elastycznego skalowania zapewnianego przez SQL Database można swobodnie przełączać się pomiędzy wydaniem Web i Business – w dowolnym kierunku. Można również przełączać zakresy rozmiarów (1 GB albo 5 GB dla wydania Web, oraz od 10 GB do 150 GB dla wydania Business) w miarę zmieniających się potrzeb. A jeśli 150 GB okaże się nadal zbyt mało, można spartycjonować bazę danych używając specjalnych technik nazywanych *fragmentowaniem* (*sharding*), co wyjaśnimy w rozdziale 8, „Projektowanie i dostrajanie na potrzeby skalowalności i wysokiej wydajności”.

Aby utworzyć nową bazę danych SQL Database, wykonaj poniższe kroki:

1. Jeśli kontynuujesz poprzednią procedurę, kliknij łącze **DATABASES** (bazy danych) u góry strony i przejdź do kroku 4. W przeciwnym razie, jeśli wylogowałeś się poprzednio i zaczniesz od początku, przejdź do kroku 2.
2. Zaloguj się w portalu Microsoft Azure pod adresem <https://manage.windowsazure.com>. Zostaniesz przekierowany do głównej strony portalu prezentującej **all items**, jak na pokazanym wcześniej rysunku 1-3.
3. Kliknij **SQL DATABASES** w pionowym panelu nawigacyjnym po lewej.
4. Kliknij łącze **CREATE A SQL DATABASE** (utwórz bazę danych), pokazane na rysunku 1-8. Pojawi się okno dialogowe **NEW SQL DATABASE**.
5. Wpisz nazwę dla nowej bazy: **WineCloudDb** (tak, będziemy zajmować się handlem winami!).
6. Pozostaw ustawienia domyślne, aby utworzyć bazę danych w wydaniu Web o wielkości do 1 GB, używającą ustawień sortowania (*collation*) **SQL_Latin1_General_CP1_CI_AS**.

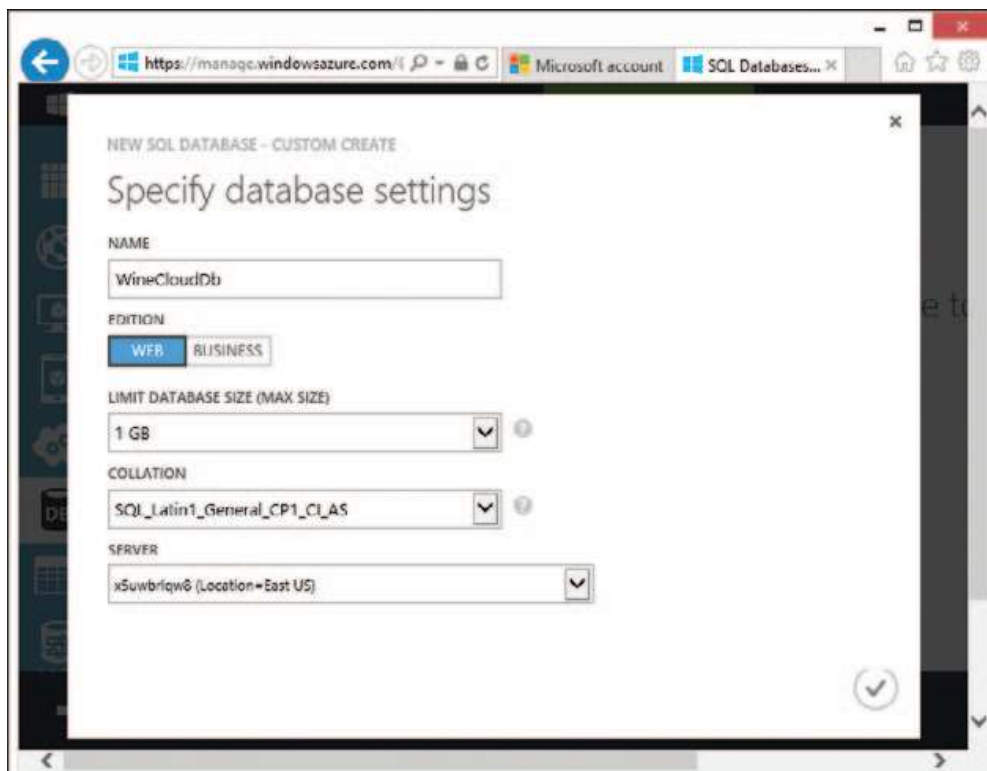


UWAGA W rozdziale 2 omówimy różnice pomiędzy wydaniem Web i Business, ustawienia maksymalnej wielkości bazy danych i znaczenie ustawień sortowania SQL Database.



RYSUNEK 1-8 Łącze CREATE A SQL DATABASE na stronie SQL DATABASES portalu

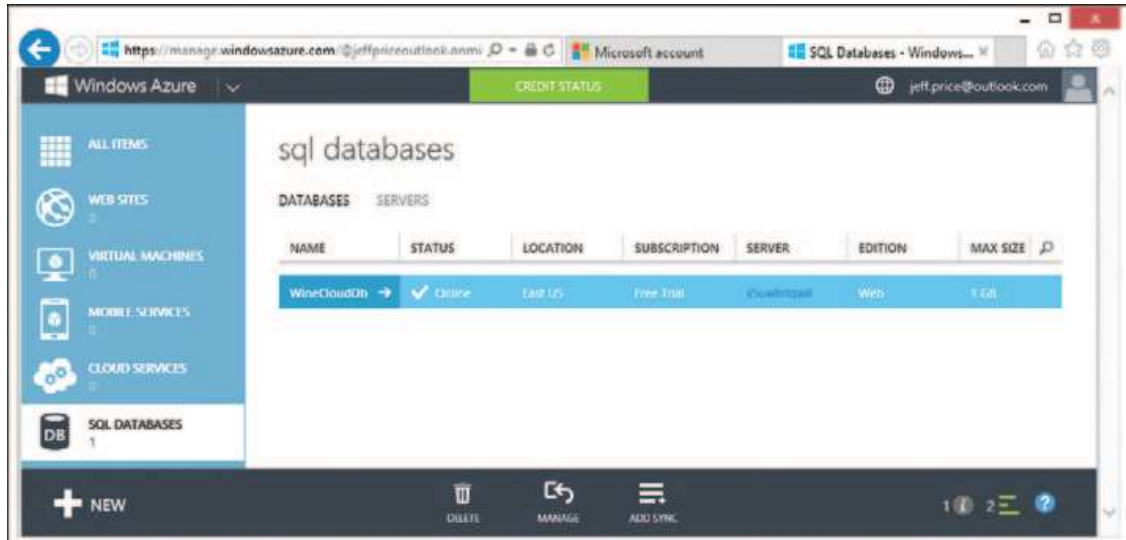
- Wybierz z listy rozwijanej serwer utworzony w poprzedniej procedurze. Strona portalu powinna wyglądać podobnie do pokazanej na rysunku 1-9.



RYSUNEK 1-9 Okno dialogowe NEW SQL DATABASE

- Kliknij ikonę zatwierdzenia w prawym dolnym rogu, aby dokończyć procedurę.

Po kilku chwilach nowa baza danych *WineCloudDb* zostanie utworzona i udostępniona do użycia, co prezentuje rysunek 1-10.



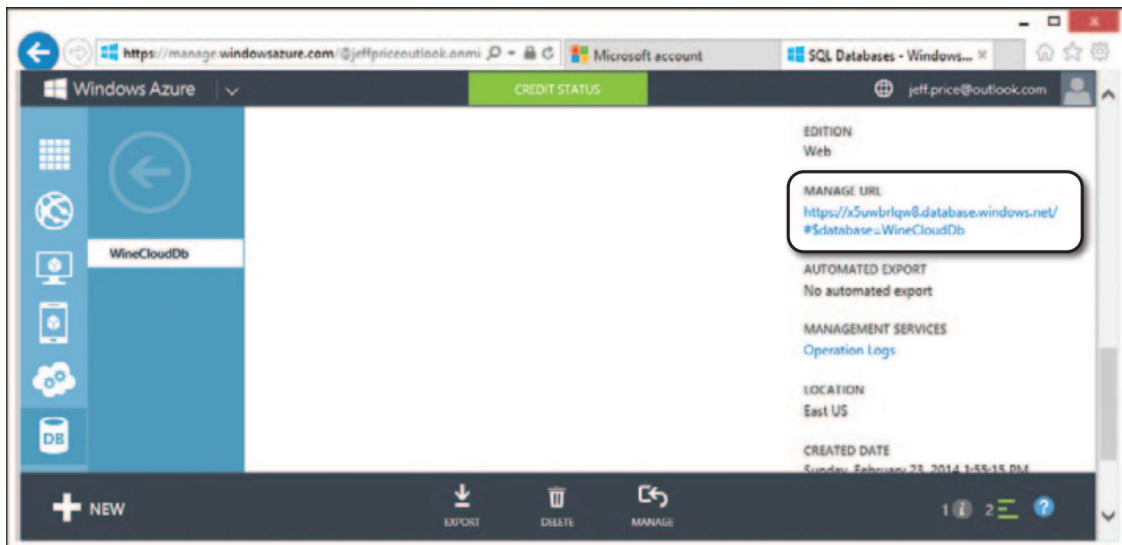
RYSUNEK 1-10 Nowa baza danych *WineCloudDb*

Korzystanie z portalu zarządzania SQL Database

Jak dotąd, posługiwaliśmy się portalem zarządzania Microsoft Azure, aby utworzyć serwer SQL Database i bazę danych. Portal ten jest interfejsem opartym na HTML, który – nawiasem mówiąc – zastąpił wcześniejszą wersję opartą na Silverlight w połowie 2012 roku. Jednak samo *projektowanie* bazy danych jest wykonywane poprzez inną stronę: portal zarządzania SQL Database, który nadal jest oparty na Silverlight (przynajmniej w czasie pisania tych słów). W tym podrozdziale pokażemy, jak dostać się do portalu zarządzania SQL Database z portalu Microsoft Azure.

Aby przejść do portalu zarządzania SQL Database, wykonaj następujące działania:

1. Jeśli kontynuujesz poprzednią procedurę, przejdź do kroku 4. W przeciwnym razie, jeśli się wylogowałeś, kontynuuj od kroku 2.
2. Zaloguj się w portalu Microsoft Azure pod adresem <https://manage.windowsazure.com>. Zostaniesz przekierowany do głównej strony portalu prezentującej all items, jak na pokazanym wcześniej rysunku 1-3.
3. Kliknij SQL DATABASES w pionowym panelu nawigacyjnym po lewej.
4. Kliknij bazę danych *WineCloudDb*.
5. Kliknij łącze DASHBOARD w górnej części strony.
6. Przewiń stronę w dół, aby odnaleźć łącze **MANAGE URL** (adres URL zarządzania) w sekcji quick glance (szybkie spojrzenie) po prawej stronie, jak na rysunku 1-11.
7. Kliknij łącze **MANAGE URL**. Zostanie otwarta nowa karta przeglądarki ze stroną logowania do portalu SQL Database.

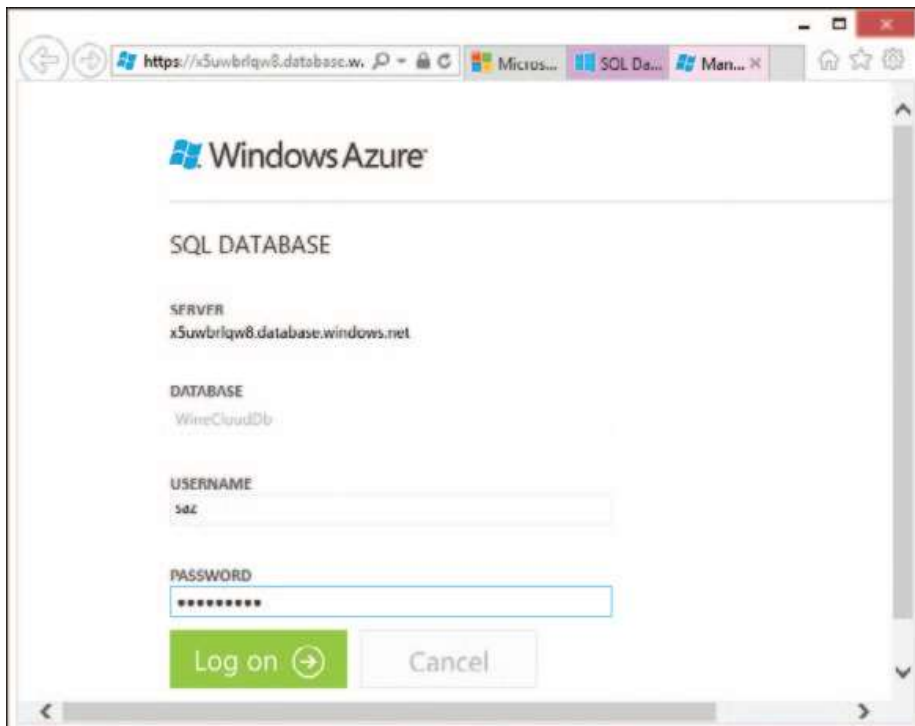


RYSUNEK 1-11 Wyszukiwanie łącza do portalu zarządzania SQL Database dla bazy danych *WineCloudDb*

UWAGA Portal SQL Database wykorzystuje technologię Silverlight. Jeśli na komputerze nie jest zainstalowane oprogramowanie Silverlight, pojawi się monit o jego pobranie i instalację, zanim będzie możliwe korzystanie z portalu.

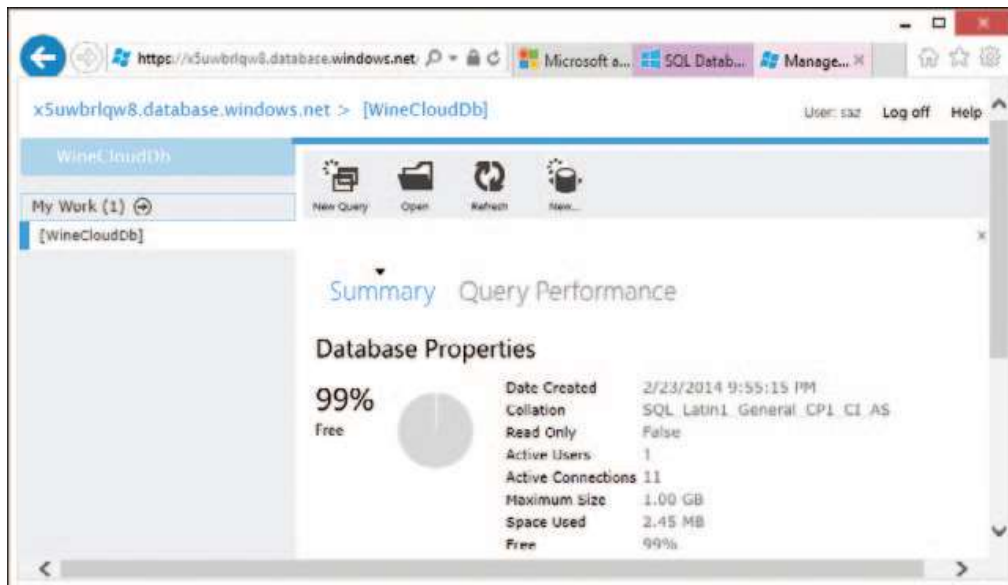


8. Wpisz nazwę użytkownika (w naszym przykładzie *saz*) oraz hasło podane przy tworzeniu serwera, jak na rysunku 1-12.



RYSUNEK 1-12 Strona logowania do portalu zarządzania SQL Database

9. Kliknij Log On. Po uwierzytelnieniu zostaniesz przekierowany do widoku Summary (podsumowanie) karty Administration dla bazy danych, pokazanego na rysunku 1-13.

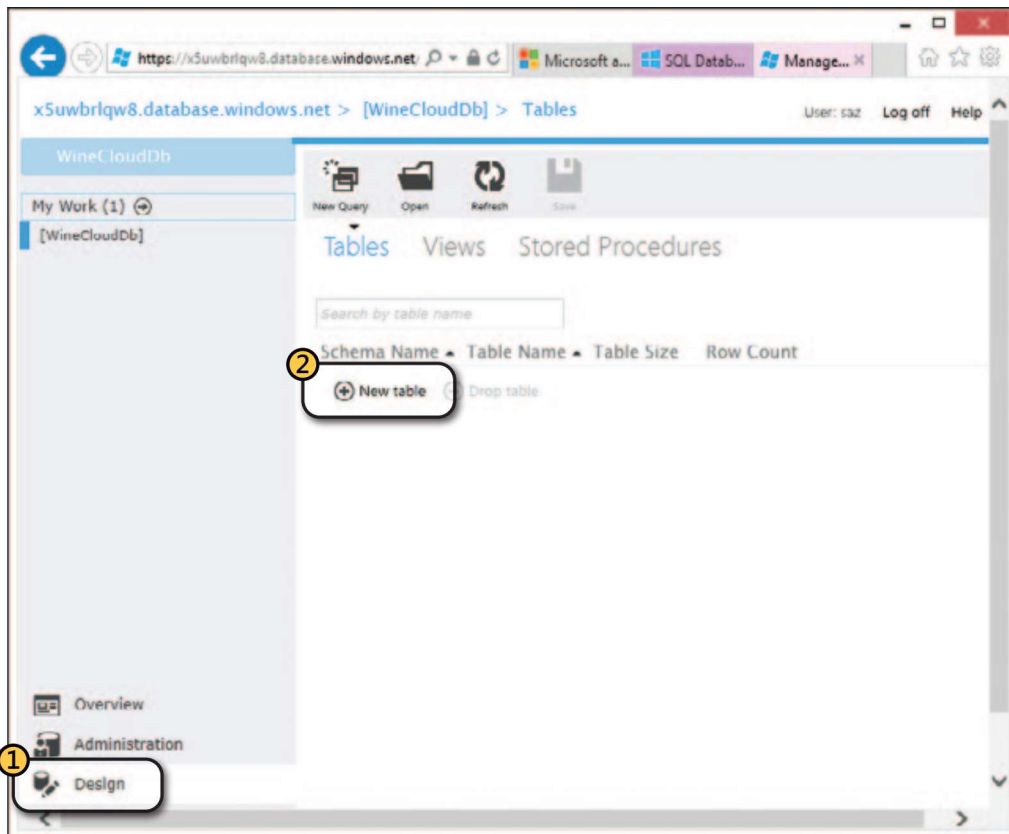


RYSUNEK 1-13 Widok Summary prezentuje właściwości bazy danych *WineCloudDb*

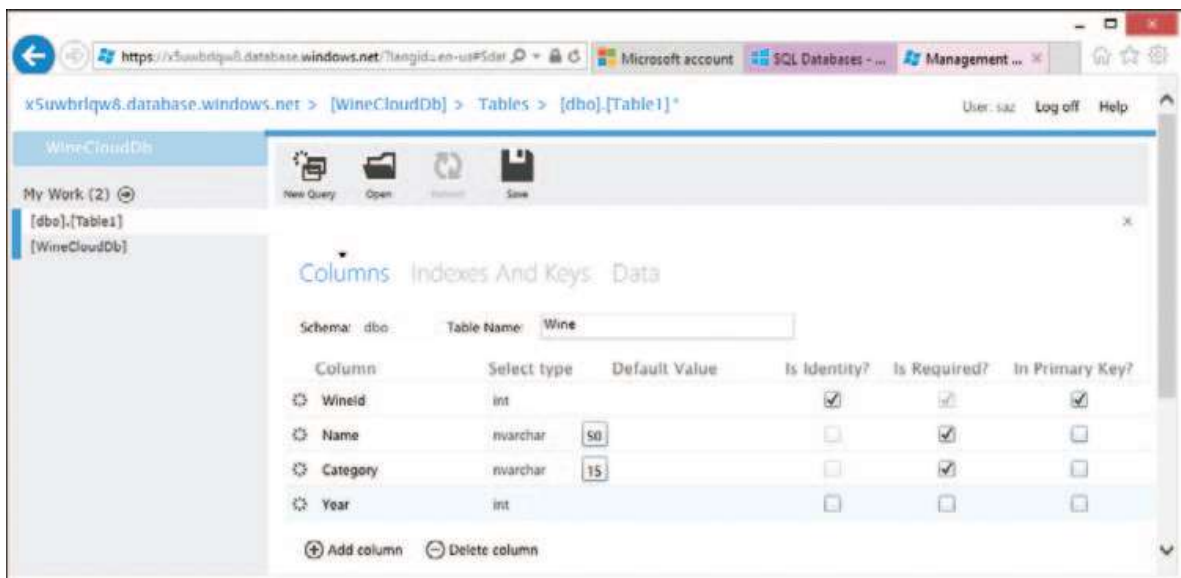
Projektowanie tabel i relacji

Możemy teraz rozpocząć tworzenie tabel. Na początek utworzymy dwie tabele, choć w dalszych rozdziałach znacznie rozszerzymy ten projekt. Tworzenie tabeli jest bardzo łatwe, co zobaczymy w kolejnej procedurze. Wystarczy po prostu kliknąć zakładkę Design (Projekt) po lewej stronie, a następnie New Table (nowa tabela), jak na rysunku 1-14.

Działanie to otworzy projektanta tabel, którego wykorzystamy do utworzenia tabeli przechowującej produkty, co pokazuje rysunek 1-15.



RYSUNEK 1-14 Tworzenie nowej tabeli



RYSUNEK 1-15 Definiowanie tabeli *Wine* przy użyciu projektanta tabel w portalu

Tworzenie tabeli Wine

Zacniemy od tworzenia tabeli *Wine* (wino). Każdy wiersz tej tabeli będzie zawierał inny gatunek wina, które nasza firma sprzedaje swoim klientom. Tabela ta będzie zawierała kolumny *WineId* (klucz główny unikatowo identyfikujący każdy produkt), *Name*, *Category* oraz *Year*.

Aby utworzyć tabelę *Wine*, wykonaj następujące działania:

1. Zaloguj się do portalu zarządzania SQL Database dla bazy danych *WineCloudDb*, zgodnie z opisem w poprzedniej procedurze.
2. Kliknij zakładkę **Design** po lewej stronie.
3. Kliknij **New Table**. Zostanie otwarty projektant tabeli z domyślną nazwą *Table1*, całkowitoliczbową kolumną *ID* oraz dwiema kolumnami tekstowymi o nazwach *Column1* i *Column2*.
4. Zmień nazwę tabeli w polu **Table Name** na *Wine*.
5. Zmień nazwę kolumny *ID* na *WineId*, pozostawiając jej ustawienie jako wymagany klucz główny (*primary key*).
6. Zaznacz pole wyboru **Is Identity?** (czy identycznościowa) dla kolumny *WineId*. Spowoduje to automatyczne przypisywanie kolejnych liczb całkowitych dla tej kolumny przez SQL Database podczas wstawiania nowych danych produktów.
7. Zmień nazwę kolumny *Column1* na *Name*, pozostawiając jej ustawienie jako wymaganej (*required*) i typ łańcucha tekstowego *nvarchar(50)*.
8. Zmień nazwę kolumny *Column2* na *Category*, pozostawiając jej ustawienie jako wymaganej i typ *nvarchar(15)*.
9. Kliknij **Add Column** (dodaj kolumnę). Spowoduje to dodanie nowej kolumny typu całkowitego (*int*) o nazwie *Column1* do projektu tabeli.
10. Zmień nazwę nowej kolumny na *Year* i pozostaw jej ustawienie jako opcjonalnej (czyli bez zaznaczenia pola wyboru **Is Required?**).
11. Kliknij **Save** w pasku narzędzi u góry okna projektanta.

Utworzyliśmy tabelę *Wine*, która powinna wyglądać analogicznie do pokazanej wcześniej na rysunku 1-15.

Tworzenie tabeli Customer

Wykonamy teraz podobną procedurę w celu utworzenia tabeli *Customer* (klient), zawierającej kolumny *CustomerId* (klucz główny), *FirstName*, *LastName* oraz *FavoriteWineId*. Kolumna *FavoriteWineId* będzie powiązana z kolumną klucza głównego *WineId* w tabeli *Wine*, zatem będzie to klucz obcy (*foreign key*) w tabeli *Customer*. Po utworzeniu tabeli *Customer* ustanowimy relację pomiędzy kolumną klucza obcego *FavoriteWineId* a kolumną klucza głównego *WineId* w tabeli *Wine*.

Aby utworzyć tabelę *Customer*, wykonaj następujące działania:

1. Kliknij zakładkę [WineCloudDb] u góry w lewym panelu strony.
2. Kliknij zakładkę **Design** u dołu po lewej. Pojawi się ta sama strona, której użyliśmy wcześniej przy tworzeniu tabeli *Wine* (rysunek 1-14).
3. Kliknij **New Table**.
4. Zmień nazwę tabeli **Customer**.
5. Zmień nazwę kolumny *ID* na **CustomerId**, pozostawiając wybór jako wymaganej i klucz główny.
6. Zaznacz pole wyboru **Is Identity?** dla kolumny *CustomerId*.
7. Zmień nazwę kolumny *Column1* na **FirstName** (imię), pozostawiając ją jako wymaganą typu *nvarchar(50)*.
8. Zmień nazwę kolumny *Column2* na **LastName** (nazwisko), pozostawiając ją jako wymaganą typu *nvarchar(15)*.
9. Kliknij **Add Column**.
10. Zmień nazwę nowej kolumny na **FavoriteWineId**, pozostawiając ją jako opcjonalną typu *int* (czyli nie zaznaczając pola wyboru **Is Required?**).
11. Kliknij **Save**.

Teraz baza danych zawiera tabele *Wine* oraz *Customer*. Tabele te będą przechowywać (co oczywiste) nasze produkty (gatunki win) oraz dane klientów, choć (co również oczywiste) obydwie są na razie puste.

Definiowanie relacji pomiędzy tabelami

Zanim przystąpimy do wypełniania tabel danymi, utworzymy relację obcego klucza pomiędzy kolumną *FavoriteWineId* w tabeli *Customer* a kolumną *WineId* w tabeli *Wine*. Warto zauważyć, że nie jest to niezbędnie wymagane w tym momencie. Tym niemniej jest zalecaną praktyką, aby poinformować SQL Database o relacjach pomiędzy tabelami, aby silnik bazy danych mógł wymusić integralność referencyjną w naszych danych. (Na przykład SQL Database będzie w stanie zagwarantować, że wartość całkowita w kolumnie *FavoriteWineId* dla każdego klienta odwołuje się do istniejącego wiersza w tabeli *Wine*, który będzie można zlokalizować poprzez *WineId*). Pozwoli to też silnikowi SQL Database na tworzenie bardziej efektywnych planów wykonywania zapytań, gdy użyjemy złączenia tabel według relacji.

Portal zarządzania SQL Database udostępnia mechanizm zarządzania kluczami obcymi, który ułatwia definiowanie relacji, pokazany na rysunku 1-16.

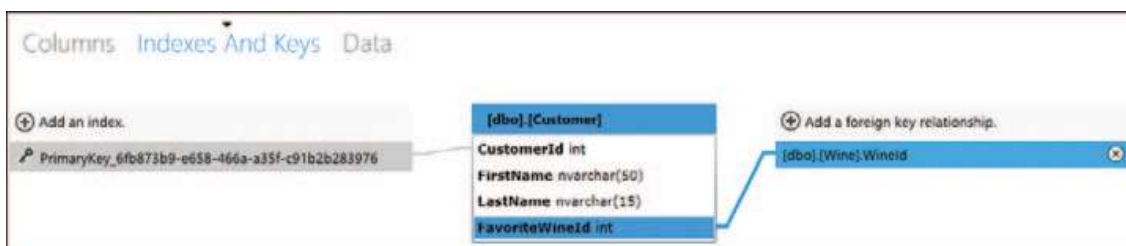


RYСУNEK 1-16 Definiowanie relacji pomiędzy tabelami przy użyciu projektanta klucza obcego

Aby zdefiniować relację pomiędzy tabelami, wykonaj następujące działania:

1. Nadal na stronie projektowania tabeli *Customer* kliknij łącze **Indexes And Keys** (indeksy i klucze) w górnej części strony.
2. Kliknij **Add A Foreign Key Relationship** (dodaj relację klucza obcego) na prawo od tabeli. Pojawi się projektant klucza obcego pokazany na rysunku 1-16.
3. Zaznacz pole wyboru przy kolumnie *FavoriteWineId* w tabeli *Customer*. Oznaczy to kolumnę jako klucz obcy.
4. Zmień nazwę relacji klucza obcego (domyślnie *FK_Customer_0*) na *FK_Customer_Wine*.
5. Kliknij **Select A Reference Table** (wybierz tabelę referencyjną) i wybierz *Wine*.
6. Kliknij **Select A Reference Column** (wybierz kolumnę referencyjną) i wybierz *WineId*.
7. Kliknij **Save**.

Relacja zostanie utworzona i kreator powinien teraz wyglądać podobnie do pokazanego na rysunku 1-17.

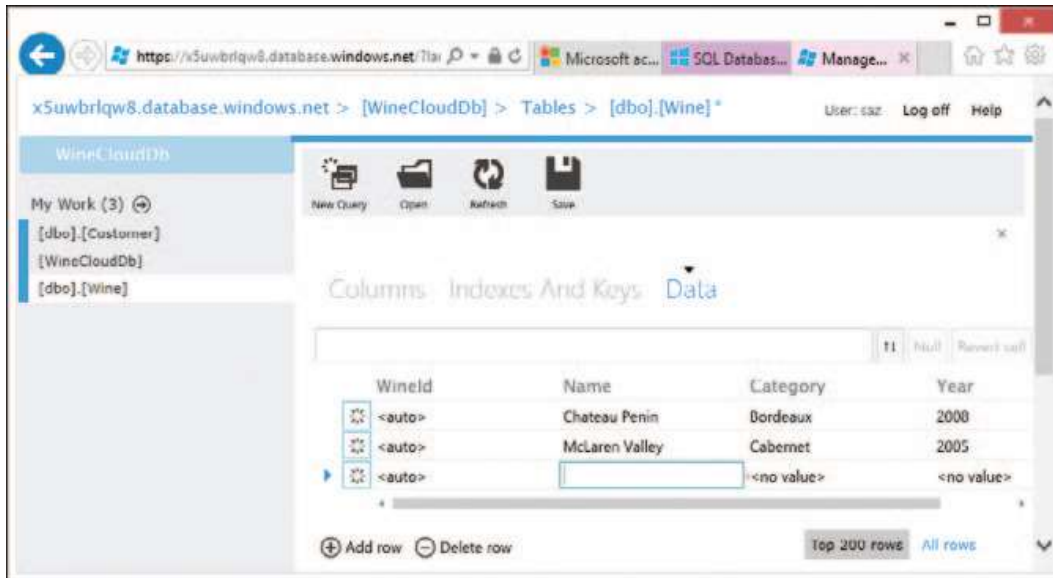


RYСУNEK 1-17 Gotowa relacja między tabelami widoczna w projektancie klucza obcego

Wstawianie danych

Najbardziej prawdopodobne scenariusze obejmują bądź migrację z istniejących baz danych SQL Server w siedzibie (co omówimy w rozdziale 4, „Migrowanie baz danych”), albo zbudowanie aplikacji, które ładują dane do bazy (czym zajmiemy się

w rozdziale 10). Jednak dla szybkiego i prostego wstawienia danych portal zarządzania SQL Database udostępnia wygodną metodę interaktywnego wprowadzania wierszy danych do tabel (co widać na rysunku 1-18), bez konieczności kodowania wyrażeń *INSERT* języka Transact-SQL (T-SQL).



RYСУNEK 1-18 Wypełnianie tabeli danymi

Aby wprowadzić do tabeli *Wine* kilka przykładowych produktów, wykonaj następujące działania:

1. Zaloguj się do portalu zarządzania SQL Database dla bazy danych *WineCloudDb* i przejdź do strony projektowania tabeli *Wine*, jak w poprzednich procedurach.
2. Kliknij łącze *Data* (dane) u góry strony.
3. Kliknij *Add Row* (dodaj wiersz).
4. Wpisz nowe wartości dla kolumn *Name*, *Category* oraz *Year*: *Chateau Penin*, *Bordeaux* i *2008*, odpowiednio.

UWAGA Ponieważ zaznaczyliśmy pole wyboru *Is Identity?* dla kolumny *Winelid* podczas projektowania tabeli, projektant wyświetla w niej wpis *<auto>* w celu zasygnalizowania, że SQL Database automatycznie wstawi tam wartość, gdy utworzone wiersze zostaną zapisane w bazie danych.



5. Powtórz kroki 3 i 4, aby wprowadzić trzy inne wina do tabeli:
 - a. McLaren Valley, Cabernet, 2005.
 - b. Mendoza, Merlot, 2010.
 - c. Valle Central, Merlot, 2009.