

Power Query w Excelu i Power BI

Zbieranie i przekształcanie danych

WYDANIE II

Daniil Maslyuk • Gil Raviv

Helion 

Tytuł oryginału: Collect, Combine, and Transform Data Using Power Query in Power BI and Excel, 2nd Edition

Tłumaczenie: Anna Mizerska, z wykorzystaniem fragmentów poprzedniego wydania w przekładzie Andrzeja Watraka

ISBN: 978-83-289-2698-1

Authorized translation from the English language edition, entitled COLLECT, COMBINE, AND TRANSFORM DATA USING POWER QUERY IN POWER BI AND EXCEL 2nd Edition by DANIIL MASLYUK; GIL RAVIV published by Pearson Education, Inc, publishing as MICROSOFT PRESS, Copyright © 2025 by Gil Raviv.

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 Helion S.A., Copyright © 2025.

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.

Helion S.A.

ul. Kościuszki 1c, 44-100 Gliwice

tel. 32 230 98 63

e-mail: helion@helion.pl

WWW: helion.pl (księgarnia internetowa, katalog książek)

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

helion.pl/user/opinie/poque2

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

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

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

Spis treści

	<i>Podziękowania</i>	13
	<i>O autorach</i>	16
	<i>Wprowadzenie</i>	17
Rozdział 1.	Wprowadzenie do Power Query	29
	Czym jest Power Query?	30
	Historia Power Query w skrócie	31
	Gdzie znajduje się Power Query?	34
	Główne komponenty Power Query	35
	Pobieranie danych i tworzenie połączeń	35
	Najważniejsze elementy edytora Power Query	37
	Różnice między starym a nowym interfejsem użytkownika edytora Power Query	41
	Ćwiczenie 1.1. Pierwsze kroki z Power Query	43
	Podsumowanie	47
Rozdział 2.	Podstawowe operacje przygotowywania danych	48
	Wyodrębnianie informacji z zakodowanych danych	49
	Fabryka Przygód	49
	Ćwiczenie 2.1. Stary sposób: formuły Excela	50
	Ćwiczenie 2.2. Część I. Nowy sposób	51
	Ćwiczenie 2.2. Część II. Scalanie tabel wyszukiwania	54
	Ćwiczenie 2.2. Część III. Tabele faktów i wyszukiwania	58

Kolumna z przykładów	60
Ćwiczenie 2.3. Część I. Kolumna z przykładów — wprowadzenie	60
Kolumna z przykładów w praktyce	62
Ćwiczenie 2.3. Część II. Zamiana wielkości na oznaczenie zakresu	63
Wyodrębnianie informacji z kolumn tekstowych	65
Ćwiczenie 2.4. Wyodrębnianie odnośników z komunikatów tekstowych	66
Operacje na danych	73
Ćwiczenie 2.5. Operacje na różnych formatach dat	73
Ćwiczenie 2.6. Operacje na danych z różnymi ustawieniami regionalnymi	74
Wyodrębnianie elementów daty i czasu	77
Przygotowanie modelu	78
Ćwiczenie 2.7. Dzielenie danych na tabelę faktów i tabelę wyszukiwania	79
Ćwiczenie 2.8. Zamiana wartości oddzielonych ogranicznikami na wiersze	81
Podsumowanie	84
Rozdział 3. Łączenie danych z kilku źródeł	85
Łączenie kilku tabel	85
Łączenie dwóch tabel	85
Ćwiczenie 3.1. Rowery i akcesoria	86
Ćwiczenie 3.2. Część I. Przekształcenie „Dołącz zapytania jako nowe”	88
Ćwiczenie 3.2. Część II. Zależności między zapytaniami i odwołania	89
Łączenie trzech lub więcej tabel	91
Ćwiczenie 3.2. Część III. Rowery + Akcesoria + Komponenty	91
Ćwiczenie 3.2. Część IV. Rowery + Akcesoria + Komponenty + Ubrania	93
Łączenie tabel na szerszą skalę	94
Łączenie tabel zapisanych w folderze	94
Ćwiczenie 3.3. Łączenie skrótych z produktami zapisanymi w folderze	95
Uwagi do importowania plików z folderu	97
Łączenie arkuszy w skoroszybie	97
Ćwiczenie 3.4. Łączenie arkuszy — rozwiązanie	98
Podsumowanie	103

Rozdział 4.	Łączenie niezgodnych tabel	104
	Problem z niezgodnymi tabelami	104
	Co to jest niezgodność tabel?	105
	Symptomy niezgodności tabel i związane z tym ryzyka	105
	Ćwiczenie 4.1. Uzgadnianie kolumn: podejście reaktywne	106
	Łączenie niezgodnych tabel zapisanych w folderze	107
	Ćwiczenie 4.2. Część I. Symptom braku wartości	107
	Ćwiczenie 4.2. Część II. Założenie takiej samej kolejności kolumn i ujednolicenie nagłówków	109
	Ćwiczenie 4.3. Proste ujednolicenie nagłówków za pomocą funkcji Table.TransformColumnNames	110
	Tabela konwersji	113
	Ćwiczenie 4.4. Transpozycja z użyciem tabeli konwersji	114
	Ćwiczenie 4.5. Anulowanie przestawienia, scalenie i powtórne przestawienie	118
	Ćwiczenie 4.6. Transpozycja samych nazw kolumn	120
	Ćwiczenie 4.7. Ujednolicenie nazw kolumn za pomocą języka M	125
	Podsumowanie	128
Rozdział 5.	Pozyskiwanie kontekstu	130
	Pozyskiwanie kontekstu zawartego w nazwach plików i skoroszytów	130
	Ćwiczenie 5.1. Część I. Kolumna niestandardowa	131
	Ćwiczenie 5.1. Część II. Pozyskiwanie kontekstu z nazwy pliku lub arkusza	132
	Pozyskiwanie kontekstu z tytułów tabel przed ich połączeniem	133
	Ćwiczenie 5.2. Pozyskiwanie kontekstu z tytułów tabel za pomocą przekształcenia Drill Down	134
	Ćwiczenie 5.3. Pozyskiwanie kontekstu z tytułów tabel podczas importowania plików z folderu	138
	Pozyskiwanie kontekstu z tytułów tabel po ich połączeniu	140
	Ćwiczenie 5.4. Pozyskiwanie kontekstu z tytułów tabel zapisanych w arkuszach jednego skoroszytu	140
	Wskazówki kontekstowe	144
	Ćwiczenie 5.5. Kolumna indeksu jako wskazówka	145
	Ćwiczenie 5.6. Wyszukiwanie informacji kontekstowych w pobliżu określonych komórek	148
	Podsumowanie	151

Rozdział 6.	Dekomponowanie tabel	153
	Rozpoznawanie źle zaprojektowanych tabel	154
	Wprowadzenie do anulowania przestawienia	156
	Ćwiczenie 6.1. Przekształcenia Anuluj przestawienie kolumn i Anuluj przestawienie innych kolumn	157
	Ćwiczenie 6.2. Anulowanie przestawienia tylko zaznaczonych kolumn	159
	Przetwarzanie sum całkowitych	161
	Ćwiczenie 6.3. Dekompozycja tabeli zawierającej sumy całkowite	161
	Dekompozycja tabeli z hierarchią agregacji danych 2×2	163
	Ćwiczenie 6.4. Dekompozycja tabeli z hierarchią agregacji danych 2×2 i datami	164
	Ćwiczenie 6.5. Dekompozycja tabeli z hierarchią agregacji danych 2×2	166
	Przetwarzanie sum częściowych	168
	Ćwiczenie 6.6. Przetwarzanie sum częściowych	169
	Podsumowanie	171
Rozdział 7.	Zaawansowane metody dekomponowania i agregowania tabel	172
	Dekomponowanie tabeli z wielopoziomową hierarchią agregacji danych	173
	Wirtualna tabela przestawna, pola wierszy i kolumn	173
	Ćwiczenie 7.1. Dekompozycja tabeli z hierarchią agregacji danych N×M	174
	Uogólnienie sekwencji przekształceń dekomponujących tabelę	177
	Ćwiczenie 7.2. Zaczynając od końca	177
	Ćwiczenie 7.3. Tworzenie funkcji FnDekomponujTabelę	178
	Przekształcenie Kolumna przestawna	187
	Ćwiczenie 7.4. Przywrócenie niewłaściwie zdekomponowanej tabeli	188
	Ćwiczenie 7.5. Agregowanie tabel z wielowierszowymi rekordami	190
	Podsumowanie	193
Rozdział 8.	Praca grupowa	195
	Pliki lokalne, parametry i szablony	196
	Niewłaściwe korzystanie z lokalnych plików	196
	Ćwiczenie 8.1. Zdefiniowanie parametru z nazwą ścieżki	197
	Ćwiczenie 8.2. Tworzenie szablonów w Power BI	199
	Ćwiczenie 8.3. Definiowanie parametrów w Excelu	200

Praca nad współdzielonymi plikami i folderami	207
Importowanie danych z plików zapisanych w usługach OneDrive dla Firm i SharePoint	207
Ćwiczenie 8.4. Tworzenie zapytań łączących się z usługami OneDrive dla Firm i SharePoint	209
Ćwiczenie 8.5. Od lokalnych folderów do usługi SharePoint	210
Zagadnienia bezpieczeństwa	213
Usuwanie zapytań w drodze inspekcji skoroszytu Excela	214
Podsumowanie	215
Rozdział 9. Wprowadzenie do języka M	216
Nauka języka M	217
Etapy opanowania języka M	217
Informacje online	220
Informacje offline	220
Ćwiczenie 9.1. Zmienna #shared i dokumentacja wbudowanych funkcji	221
Bloki konstrukcyjne w języku M	222
Ćwiczenie 9.2. Witaj, świecie!	222
Wyrażenie let	224
Zakresy i scalanie wyrażeń z wielu zapytań	225
Typy danych, operatory i wbudowane funkcje	228
Podstawowe typy danych w języku M	230
Typ number	231
Typ time	231
Typ date	232
Typ duration	233
Typ text	233
Typ null	234
Typ logical	235
Typy złożone	235
Typ list	236
Typ record	239
Typ table	242
Warunki i wyrażenia „if”	243

Wyrażenie „if-then-else”	244
Wyrażenie if wewnątrz wyrażenia let	245
Funkcje niestandardowe	246
Wywoływanie funkcji	248
Wyrażenie each	249
Zaawansowane tematy	249
Obsługa błędów	250
Leniwe i zachłanne przetwarzanie wyrażań	251
Pętle	251
Rekurencja	252
Funkcja List.Generate	253
Funkcja List.Accumulate	253
Podsumowanie	255
Rozdział 10. Od złych do dobrych zapytań	256
Źródła pułapek i ich skutki	257
Świadome działanie	258
Dobre praktyki	259
Modyfikacje formuł M	259
Pułapka 1. Ignorowanie paska formuły	260
Ćwiczenie 10.1. Wyszukiwanie w pasku formuły statycznych odwołań do nazw kolumn	260
Pułapka 2. Zmiana typów danych	262
Pułapka 3. Niebezpieczne filtrowanie	265
Ćwiczenie 10.2. Część I. Filtrowanie czarnych produktów	265
Działanie warunków w filtrach	267
Ćwiczenie 10.2. Część II. Wyszukiwanie wartości w panelu filtru	268
Pułapka 4. Zmiana kolejności kolumn	269
Ćwiczenie 10.3. Część I. Zmiana kolejności wybranych kolumn	269
Ćwiczenie 10.3. Część II. Niestandardowa funkcja FnZmianaKolejnościKolumn	271
Pułapka 5. Usuwanie i wybieranie kolumn	272
Ćwiczenie 10.4. Usunięcie niepotrzebnych kolumn	272
Pułapka 6. Zmiana nazw kolumn	274
Ćwiczenie 10.5. Zmiana nazw losowych kolumn	275

Pułapka 7. Dzielenie kolumn	277
Ćwiczenie 10.6. Niewłaściwy podział kolumny	278
Pułapka 8. Scalanie kolumn	281
Inne pułapki i techniki zapobiegawcze	282
Podsumowanie	282
Rozdział 11. Profilowanie danych	285
Sprawdzanie jakości kolumn	285
Ćwiczenie 11.1. Część I. Sprawdzanie i usuwanie błędów	285
Ćwiczenie 11.1. Część II. Usuwanie pustych wartości	287
Przeglądanie szczegółów profilu kolumn	288
Ćwiczenie 11.2. Część I. Wyświetlanie szczegółów profilu kolumny	288
Ćwiczenie 11.2. Część II. Wyświetlanie wartości odrębnych i unikatowych	289
Używanie funkcji Table.Schema i Table.Profile	291
Ćwiczenie 11.3. Zastosowanie Table.Schema i Table.Profile	291
Podsumowanie	294
Rozdział 12. Podstawy analizy tekstu	295
Wyszukiwanie słów kluczowych w kolumnach tekstowych	296
Ćwiczenie 12.1. Proste wyszukiwanie słów kluczowych	296
Wyszukiwanie słów kluczowych z użyciem iloczynu kartezjańskiego	299
Ćwiczenie 12.2. Iloczyn kartezjański	301
Ćwiczenie 12.3. Wyszukiwanie słów kluczowych za pomocą niestandardowej funkcji	307
Jakiej metody użyć: zwykłego wyszukiwania, iloczynu kartezjańskiego czy niestandardowej funkcji?	310
Dzielenie wpisów na słowa	310
Ćwiczenie 12.4. Trywialne dzielenie wpisów na słowa	311
Ćwiczenie 12.5. Odrzucenie stop-słów	315
Ćwiczenie 12.6. Wyszukiwanie słów kluczowych w wyodrębnionych słowach	318
Ćwiczenie 12.7. Chmura słowna w Power BI Desktop	325
Podsumowanie	327

Rozdział 13.	Zaawansowana analiza tekstu: rozpoznawanie znaczenia	328
	Usługa Microsoft Azure AI Services	328
	Klucze API i zasoby Azure	331
	Uwagi dotyczące korzystania z usługi Azure AI Services za pomocą edytora Power Query	334
	Tłumaczenie tekstu	336
	Opis interfejsu Text Translation API	336
	Ćwiczenie 13.1. Proste tłumaczenie	337
	Ćwiczenie 13.2. Tłumaczenie wielu komunikatów	342
	Analiza nastroju tekstu	346
	Czym jest interfejs Sentiment Analytics API	347
	Ćwiczenie 13.3. Implementacja niestandardowej funkcji FnOkreślenieNastroju	349
	Ćwiczenie 13.4. Przetwarzanie dużych ilości danych	359
	Wyodrębnianie kluczowych fraz	361
	Ćwiczenie 13.5. Przystosowanie kodu do wyodrębniania kluczowych fraz	361
	Obsługa wielu języków	364
	Zmiana oznaczenia języka	364
	Dynamiczne wykrywanie języka	364
	Ćwiczenie 13.6. Przystosowanie kodu do wykrywania języka	365
	Podsumowanie	366
Rozdział 14.	Diagnostyka zapytań	367
	Włączanie diagnostyki zapytań	367
	Korzystanie z diagnostyki zapytań	369
	Składanie zapytań	370
	Ćwiczenie 14.1. Używanie diagnostyki zapytań do sprawdzania zwijania zapytań	372
	Podsumowanie	375
Rozdział 15.	Power Query Online	376
	Praca z Power Query Online	376
	Różnice między Power Query dla Desktop a Power Query Online	378
	Puste wartości kontra wartości null	378
	Typy danych	379

Kiedy używać przepływów danych?	380
Centralizacja logiki	380
Poprawa wydajności zapytań	380
Zmniejszenie zużycia zasobów	381
Niezależna współpraca	382
Ćwiczenie 15.1. Tworzenie przepływu danych	382
Pobieranie danych z przepływów danych	383
Ćwiczenie 15.2. Część I. Tworzenie funkcji wybierania tabeli przepływu danych	385
Ćwiczenie 15.2. Część II. Zastępowanie pustych wartości wartością null we wszystkich kolumnach tekstowych	387
Podsumowanie	388
 Rozdział 16. Końcowy projekt: wszystko razem	390
Ćwiczenie 16.1. Ratowanie sytuacji u Światowych Importerów	390
Wskazówki	391
Ćwiczenie 16.1. Część I. Przygotowanie danych	392
Ćwiczenie 16.1. Część II. Dekompozycja tabel	394
Ćwiczenie 16.1. Część III. Agregacja danych za 2025 r.	394
Ćwiczenie 16.1. Część IV. Scalenie przychodów za lata 2022 – 2024 i za rok 2025	396
Ćwiczenie 16.2. Porównanie tabel i śledzenie hakera	396
Wskazówki	397
Ćwiczenie 16.2. Część I. Znajdowanie niespójności w przychodach	397
Ćwiczenie 16.2. Część II. Szukanie śladów hakera w zmienionej tabeli	399
Podsumowanie	400

ROZDZIAŁ 11.

Profilowanie danych

Możliwe, że przed przystąpieniem do analizy będziesz chciał zobaczyć ogólny kształt lub strukturę danych, a w tym pomoże Ci profilowanie danych. **Profilowanie danych** odnosi się do technicznej analizy danych mającej na celu ocenę ich zawartości i spójności. Z profilowania danych warto skorzystać przed ich wizualizacją, ponieważ możesz odkryć pewne ograniczenia lub nieoczekiwane wartości w swoich danych. Bez profilowania danych możesz przeoczyć wartości odstające, które mogą zniekształcić Twoją analizę.

Do profilowania danych możesz używać różnych narzędzi, w tym Power Query, co oznacza, że możesz profilować swoje dane przed załadowaniem ich do modelu danych. W interfejsie użytkownika zobaczysz jakość kolumn, rozkład wartości i profil, a także możesz użyć wybranych funkcji języka formuł Power Query do bardziej zaawansowanego profilowania.

W TYM ROZDZIALE NAUCZYSZ SIĘ:

- różnych sposobów profilowania danych,
- sprawdzać jakość kolumn, rozkład i profil,
- używać funkcji `Table.Schema` do opisywania tabeli,
- profilować tabelę za pomocą funkcji `Table.Profile`.

Sprawdzanie jakości kolumn

W kontekście profilowania danych w Power Query jakość kolumny odnosi się do proporcji wartości prawidłowych, błędnych i pustych. Poprzez sprawdzenie jakości kolumn możesz rozwiązywać problemy z błędnymi wartościami przed załadowaniem danych, a także upewnić się, czy nie ma pustych wartości, jeśli nie spodziewasz się ich w swoich danych.

Ćwiczenie 11.1. Część I. Sprawdzanie i usuwanie błędów

W tym rozdziale będziesz korzystać z pliku *R11-01.csv* jako danych źródłowych we wszystkich ćwiczeniach. W tym ćwiczeniu użyjesz profilowania kolumn w Power Query, aby sprawdzić błędy w kolumnie.

1. Otwórz edytor Power Query i pobierz dane z pliku *R11-01.csv*, akceptując domyślne ustawienia pliku.
2. Włącz widok jakości kolumn w karcie *Widok*:
 - W nowym interfejsie użytkownika Power Query, w menu rozwijanym *Widok danych*, zaznacz *Włącz profil kolumny* i *Pokaż szczegóły jakości kolumn*.
 - W starym interfejsie użytkownika Power Query, w grupie *Podgląd danych*, zaznacz *Jakość kolumn*.
3. Zauważ, że wszystkie kolumny pokazują 100% prawidłowych wartości, co widać na rysunku 11.1.

	Miesiąc	Region	Kraj	Nazwa produktu	Rozmiar	1.2 Całkowita kwota	Całkowita ilość	1.2 Średnia cena
	• Prawidłowe 100% • Błąd 0% • Puste 0%	• Prawidłowe 100% • Błąd 0% • Puste 0%	• Prawidłowe 100% • Błąd 0% • Puste 0%	• Prawidłowe 100% • Błąd 0% • Puste 0%	• Prawidłowe 100% • Błąd 0% • Puste 0%	• Prawidłowe 100% • Błąd 0% • Puste 0%	• Prawidłowe 100% • Błąd 0% • Puste 0%	• Prawidłowe 100% • Błąd 0% • Puste 0%
1	1.06.2002	Europa	Niemcy	Rower szosowy-250 Czarny, 48	48	2443,35	2	2443,35
2	1.06.2002	Północna Ameryka	Stany Zjednoczone	Rower szosowy-250 Czarny, 48	48	2443,35	2	2443,35
3	1.06.2002	Północna Ameryka	Stany Zjednoczone	Rower szosowy-250 Czerwony, 58	58	2443,35	2	2443,35
4	1.06.2002	Pacyfik	Australia	Rower szosowy-250 Czarny, 48	48	2443,35	2	2443,35
5	1.06.2002	Pacyfik	Australia	Rower szosowy-250 Czerwony, 58	58	2443,35	2	2443,35
6	1.07.2002	Północna Ameryka	Stany Zjednoczone	Rower szosowy-250 Czarny, 44	44	2443,35	2	2443,35

RYСУNEK 11.1. Dla każdej kolumny podano procent wartości prawidłowych, błędnych i pustych

4. W dolnej części okna edytora Power Query wybierz *Profilowanie kolumn w oparciu o następującą liczbę pierwszych wierszy: 1000*, a następnie przełącz na opcję *Profilowanie kolumn w oparciu o cały zestaw danych*.
5. Zwróć uwagę, że teraz kolumna *Rozmiar* ma 32% wartości błędnych. Jest to spowodowane błędami występującymi w wierszach po wierszu 1000. Przeanalizujmy źródło błędów.
6. Zaznacz kolumnę *Rozmiar* i w karcie *Narzędzia główne* wybierz *Zachowaj wiersze*, a potem *Zachowaj błędy*.
7. Wybierz jedną z komórek z wartością Error. Powinieneś zobaczyć komunikat o błędzie podobny do tego:
 Nie możemy przekonwertować na typ Number.
 Ten komunikat o błędzie oznacza, że Power Query próbował przekonwertować niektóre wartości tekstowe na liczby. Ponieważ wartości w kolumnie *Rozmiar* mogą być zarówno liczbowe, na przykład 48, jak i tekstowe, na przykład M, typ kolumny musi być tekstowy, a nie liczbowy, tak jak Power Query wykryło automatycznie.
8. Z listy *Zastosowane kroki* usuń krok *Zachowano błędy* i zmień typ kolumny *Rozmiar* na tekst. W tym celu kliknij prawym przyciskiem nagłówek kolumny *Rozmiar* i wybierz *Zmień typ*, a następnie *Tekst*. Gdy zostaniesz zapytany, czy chcesz zastąpić bieżący krok, czy dodać nowy, wybierz *Zastąp bieżący*. Ważne jest, aby nie dodawać nowego kroku, ponieważ zmiana typu błędnych wartości nie rozwiąże problemu.

Na tym etapie możesz zauważyć, że błędy z kolumny *Rozmiar* zniknęły, ale możesz również zauważyć, że w kolumnie *Rozmiar*, jak i w trzech ostatnich kolumnach, znajdują się puste wartości. W następnej części ćwiczenia przyjrzymy się tym pustym wartościom.



Uwaga. W niektórych środowiskach Power Query może być konieczne przejście do innego kroku na liście *Zastosowane kroki*, a następnie ponowne wybranie kroku *Zmieniono typ*, aby zobaczyć, że błędy zniknęły.

Ćwiczenie 11.1. Część II. Usuwanie pustych wartości

W części II ćwiczenia 11.1 sprawdzisz puste wartości w czterech ostatnich kolumnach. Czasami puste wartości oznaczają brakujące dane, a czasami puste wartości są uzasadnione. Sprawdzenie pustych wartości pozwala zdecydować, czy wartości są uzasadnione, czy też warto porozmawiać z twórcą danych.

Rozpocznij tam, gdzie zakończyłeś pierwszą część tego ćwiczenia.

9. Naciśnij przycisk filtrowania w nagłówku kolumny *Rozmiar* i kliknij *Łańcuch więcej*, jeśli zobaczysz komunikat *Lista może być niekompletna*.
10. Usuń zaznaczenie wszystkich wartości, a następnie wybierz (*puste*) i kliknij *OK*.
11. Sprawdź wartości w kolumnie *Nazwa produktu*. W tym celu w nagłówku tej kolumny naciśnij przycisk filtrowania. Jak widać, w tej kolumnie jest tylko jedna wartość: *Czapka z logo AWC*. Kliknij *Anuluj*, aby zamknąć okno filtra kolumny *Nazwa produktu*.
12. Decydujesz, że zamiast pustej wartości bardziej uzasadnione jest wyświetlanie wartości *Rozmiar uniwersalny*. Kliknij prawym przyciskiem myszy pustą komórkę w kolumnie *Rozmiar* i wybierz *Zamień wartości*.
13. W polu *Zamień na* wpisz *Rozmiar uniwersalny* i naciśnij *OK*. Upewnij się, że pole *Wartość do znalezienia* jest puste.
14. Z listy *Zastosowane kroki* usuń krok *Przefiltrowano wiersze*. Gdy zostaniesz poproszony o potwierdzenie, wybierz *Usuń*.
15. Teraz kolumna *Rozmiar* ma 100% prawidłowych wartości, podczas gdy w trzech ostatnich kolumnach nadal są puste wartości. Wybierz przycisk filtrowania w nagłówku kolumny *Kwota całkowita* i kliknij *Łańcuch więcej*.
16. Usuń zaznaczenie wszystkich wartości i zaznacz (*null*), a następnie naciśnij *OK*.

Zauważ, że jest tylko jeden wiersz, w którym wartości w kolumnach *Kwota całkowita*, *Całkowita ilość* i *Średnia cena* są równe null. Jako użytkownik tego zestawu danych możesz zapisać sobie przypomnienie, aby porozmawiać z zespołem odpowiedzialnym za przygotowanie danych i zapytać, czy wartości null są prawidłowymi wpisami.



Uwaga. Jak widziałeś w tym ćwiczeniu, Power Query domyślnie wykonuje profilowanie danych jedynie na podstawie pierwszych 1000 wierszy. Chociaż idealnie byłoby profilować cały zestaw danych, należy to robić ostrożnie w przypadku dużych zbiorów danych.

Gotowe rozwiązanie jest zapisane w plikach *R11-01 - rozwiązanie.xlsx* oraz *R11-01 - rozwiązanie.pbix*.

Przeglądanie szczegółów profilu kolumn

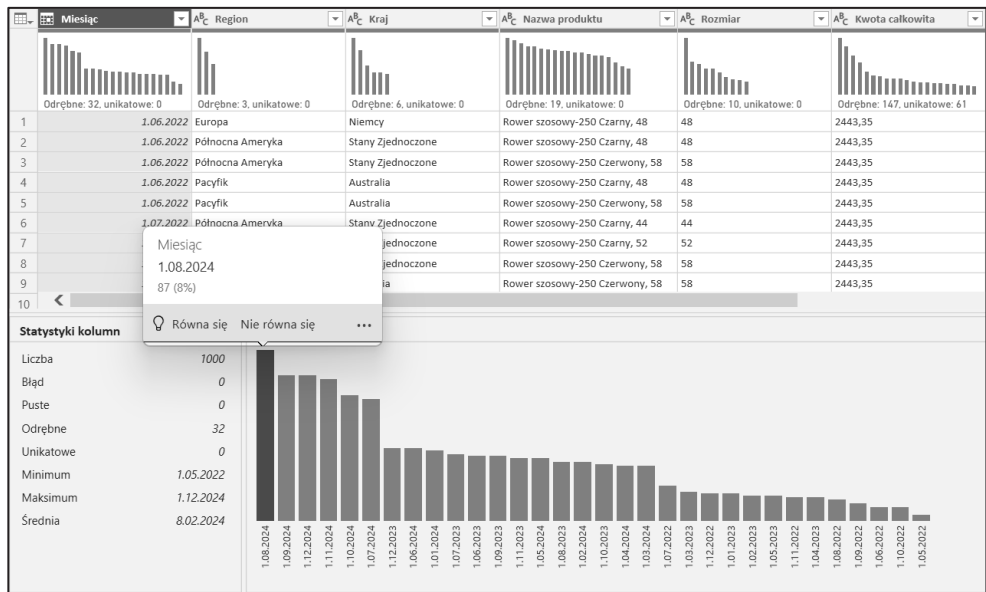
Profilowanie danych to nie tylko sprawdzanie jakości kolumn; możesz być zainteresowany rozkładem wartości, czyli zobaczeniem, jak jednolite lub różne są wartości w kolumnie, jakie rodzaje wartości są zawarte w kolumnie itp. Możesz być również zainteresowany podstawowymi statystykami, takimi jak minimalne i maksymalne wartości, liczba unikatowych wartości i tak dalej.

W interfejsie użytkownika edytora Power Query można zobaczyć szczegóły profilu kolumny. Jak się przekonasz w następnym ćwiczeniu, możesz przeglądać statystyki kolumny oraz szczegółowy rozkład wartości.

Ćwiczenie 11.2. Część I. Wyświetlanie szczegółów profilu kolumny

W tym ćwiczeniu nauczysz się, jak przeglądać statystyki kolumn oraz względną popularność różnych punktów danych w szczegółach profilu kolumny.

1. Otwórz edytor Power Query i pobierz dane z pliku *R11-01.csv*, podobnie jak w ćwiczeniu 11.1.
2. Gdy otworzy się okno z ustawieniami pliku źródłowego, z menu rozwijanego *Wykrywanie typu danych* wybierz opcję *Na podstawie całego zestawu danych*, a następnie kliknij OK, by przejść do przekształcania danych.
3. Zmień nazwę nowo utworzonego zapytania na *Sprzedaż*.
4. Włącz widok rozkładu i profilu kolumny w karcie *Widok*:
 - W nowym interfejsie użytkownika Power Query na liście rozwijanej *Widok danych* zaznacz *Włącz profil kolumny* i *Pokaż dystrybucję wartości kolumn* oraz *Pokaż profil kolumny w okienku szczegółów*.
 - W starym interfejsie użytkownika Power Query zaznacz *Rozkład kolumn* i *Profil kolumny*.
5. Jeśli to konieczne, usuń zaznaczenie szczegółów jakości kolumny w karcie *Widok*.
6. Upewnij się, że na dole edytora Power Query wybrana jest opcja *Profilowanie kolumn w oparciu o cały zestaw danych*.
7. Na tym etapie pod każdym nagłówkiem kolumny powinieneś zobaczyć wykres kolumnowy oraz liczbę wartości unikatowych i odrębnych w kolumnie. **Wartość unikatowa** to taka, która pojawia się tylko raz w kolumnie. Wykres pod nagłówkiem kolumny może dać Ci wyobrażenie o rozkładzie wartości, jednak nie dostarczy Ci szczegółów, takich jak to, która wartość jest najczęstsza w kolumnie. Jeśli chcesz zobaczyć więcej szczegółów dotyczących rozkładu wartości i statystyk kolumny, zaznacz kolumnę poprzez kliknięcie nagłówka. Możesz najechać kursorem na kolumnę w szczegółowym wykresie rozkładu wartości, aby zobaczyć, ile razy dana wartość pojawia się w kolumnie. Rysunek 11.2 przedstawia szczegóły profilu kolumny *Miesiąc*.



RYСУNEK 11.2. Szczegóły profilu kolumny pokazują statystyki kolumny i rozkład dla wybranej kolumny

- Zauważ, że teraz możesz również zobaczyć statystyki kolumny, takie jak liczba pustych oraz unikatowych wartości, wartości minimalne, maksymalne, średnie i inne. Możesz również zobaczyć bardziej szczegółowy wykres rozkładu wartości. Możesz najechać kursorem na kolumnę i zobaczyć, jak często dana wartość pojawia się w wybranej kolumnie. Jeśli chcesz, możesz pogrupować wartości inaczej: na przykład w przypadku kolumny *Miesiąc* możesz kliknąć *Więcej* (trzy kropki w prawym górnym rogu wykresu rozkładu wartości) i wybrać inny sposób grupowania wartości, na przykład według roku lub miesiąca.
- Na wykresie rozkładu wartości kliknij *Więcej*, a następnie *Grupuj według* i wybierz *Miesiąc*. Zauważ, że w tej kolumnie najczęściej pojawia się grudzień; 572 wartości z 4389.

Ćwiczenie 11.2. Część II. Wyświetlanie wartości odrębnych i unikatowych

W części II ćwiczenia 11.2 wypiszesz wartości odrębne i unikatowe z kolumny *Całkowita ilość*, które są jednymi z podstawowych sposobów profilowania danych. Warto zobaczyć wartości odrębne i unikatowe, gdy oczekujesz lub nie oczekujesz, że niektóre wartości się pojawią.

Zaczyniesz tam, gdzie skończyłeś w części 1. tego ćwiczenia.

- Kliknij prawym przyciskiem myszy zapytanie *Sprzedaż* i wybierz *Odwołanie*. Zmień nazwę nowego zapytania na *Odrębna ilość*.
- W zapytaniu *Odrębna ilość* kliknij prawym przyciskiem myszy nagłówek kolumny *Całkowita ilość* i wybierz *Usuń inne kolumny*.

12. Kliknij prawym przyciskiem myszy nagłówek kolumny *Całkowita ilość* i wybierz *Usuń duplikaty*. Na tym etapie powinieneś zobaczyć 837 wierszy, a rozkład wartości kolumny powinien wskazywać *837 odrębnych*, *837 unikatowych*. Ponieważ usunąłeś duplikaty, wszystkie wartości stały się unikatowe.
13. Kliknij prawym przyciskiem myszy zapytanie *Odrębna ilość* i wybierz *Duplikuj*. Zmień nazwę nowego zapytania na *Unikatowa ilość*.
14. W zapytaniu *Unikatowa ilość* usuń ostatni krok, *Usunięto duplikaty*.
15. Na karcie *Narzędzia główne* wybierz *Zachowaj wiersze*, a następnie *Zachowaj duplikaty*. W tym momencie powinieneś zobaczyć następujące wyrażenie na pasku formuły (lekko sformatowane dla czytelności), gdzie *#"Usunięto inne kolumny"* jest nazwą poprzedniego kroku:

```
= let
    columnNames = {"Kwota całkowita"},
    addCount = Table.Group(#"Usunięto inne kolumny", columnNames, {"Count",
        ↪Table.RowCount, type number})),
    selectDuplicates = Table.SelectRows(addCount, each [Count] > 1),
    removeCount = Table.RemoveColumns(selectDuplicates, "Count")
in
    Table.Join(#"Usunięto inne kolumny", columnNames, removeCount, columnNames,
        ↪JoinKind.Inner)
```

Zatrzymaj się na chwilę, aby zrozumieć ten kod, który Power Query napisał za Ciebie.

- Zmienna *columnNames* przechowuje nazwy kolumn, które zostaną użyte później w zmiennej *addCount* oraz w wyniku kroku po *in*.
- Zmienna *addCount* grupuje wszystkie wartości i dodaje licznik wierszy do każdej grupy w kolumnie *Count*.
- *selectDuplicates* zachowuje tylko te wartości, które pojawiają się więcej niż raz, ponieważ ten kod został wygenerowany po naciśnięciu *Zachowaj duplikaty*.
- Zmienna *removeCount* usuwa kolumnę *Count*, ponieważ nie jest już potrzebna.
- W końcu, po *in*, *Table.Join* jest używane do przefiltrowania poprzedniego kroku, aby zachować tylko te wartości, które pojawiają się więcej niż raz, w celu zachowania duplikatów.
- Jak omówiono w rozdziale 9, „Wprowadzenie do języka M”, najlepiej jest używać ostatniej zmiennej z bloku *let* zamiast wyrażenia po *in*. Jednakże, ponieważ jest to zagnieżdżone wyrażenie *let/in*, nie ma to większego znaczenia, gdyż i tak nie zobaczymy pośrednich kroków.

Warto zwrócić uwagę, jak można używać funkcji *Table.Join* do filtrowania bez konieczności usuwania dodatkowej kolumny, którą stworzyłaby funkcja *Table.NestedJoin*.

16. Unikatowe wartości są bezpośrednimi przeciwieństwami duplikatów — pojawiają się tylko raz. W pasku formuły zmień operator porównania w zmiennej *selectDuplicates* na równość zamiast znaku większości, jak pokazano pogrubionym drukiem poniżej:

```
= let
    columnNames = {"Kwota całkowita"},
```

```

addCount = Table.Group("#Usunięto inne kolumny", columnNames, {"Count",
↳Table.RowCount, type number})),
selectDuplicates = Table.SelectRows(addCount, each [Count] = 1),
removeCount = Table.RemoveColumns(selectDuplicates, "Count")
in
Table.Join("#Usunięto inne kolumny", columnNames, removeCount, columnNames,
↳JoinKind.Inner)

```

Powinieneś teraz zobaczyć 445 wierszy. Ostatnia wartość, pusta, która potencjalnie może być nieoczekiwana, jest teraz łatwiejsza do zauważenia.



Uwaga. W tym ćwiczeniu, gdy połączyliśmy się z plikiem CSV, wykryliśmy typy kolumn na podstawie całego zestawu danych. Podobnie jak w przypadku profilowania kolumn na podstawie całego zestawu danych, tu również należy używać tej opcji z ostrożnością.

Gotowe rozwiązanie jest zapisane w plikach *R11-02 - rozwiązanie.xlsx* oraz *R11-02 - rozwiązanie.pbix*.

Używanie funkcji Table.Schema i Table.Profile

Oprócz możliwości profilowania dostępnych w interfejsie użytkownika istnieją dwie funkcje, które mogą być przydatne: `Table.Schema` i `Table.Profile`.

Funkcja `Table.Schema` opisuje kolumny określonej tabeli. Listuje kolumny wraz z ich pozycjami, typami danych, informacją, czy kolumny mogą mieć puste wartości, i tak dalej. Większość kolumn zwracanych przez funkcję `Table.Schema` jest istotna tylko dla określonych źródeł danych, takich jak bazy danych SQL, chociaż typy danych kolumn są dostępne dla wszystkich źródeł danych.

Przykładem, w którym funkcja `Table.Schema` może być użyteczna, jest zmiana źródeł danych — na przykład z plików na bazę danych — i chęć upewnienia się, że stare i nowe tabele mają taką samą strukturę. Funkcji `Table.Schema` można użyć na obu tabelach, a następnie połączyć wyniki na podstawie nazw kolumn, aby zobaczyć, czy istnieją jakieś istotne różnice.

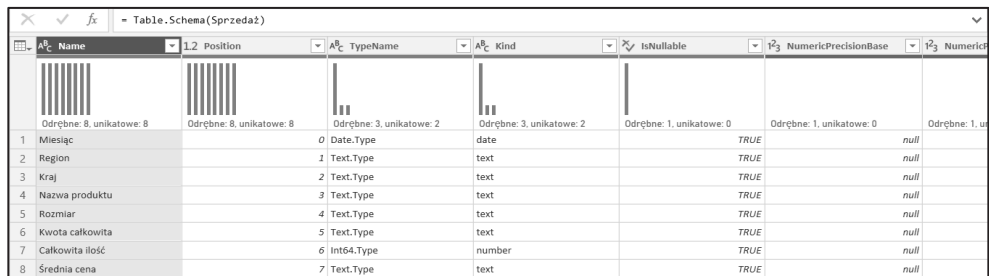
Funkcja `Table.Profile` zwraca informacje podobne do widoku statystyk kolumn dla całej tabeli. Dodatkowo można dodać swoje własne agregacje, aby zobaczyć więcej statystyk, w zależności od sposobu, w jaki chce się profilować dane.

Ćwiczenie 11.3. Zastosowanie Table.Schema i Table.Profile

W tym ćwiczeniu wykorzystasz funkcje `Table.Schema` i `Table.Profile`, aby uzyskać dodatkowe informacje o danych.

1. Otwórz edytor Power Query i zaimportuj dane z pliku *R11-01.csv*, tego samego, który pobrałeś w ćwiczeniu 11.1.
2. W oknie ustawień pliku wybierz z menu rozwijanego *Wykrywanie typu danych* opcję *Na podstawie całego zestawu danych*, a następnie kliknij OK, by przejść do przekształcania danych.

3. Zmień nazwę nowo utworzonego zapytania na *Sprzedaż*.
4. Kliknij prawym przyciskiem myszy zapytanie *Sprzedaż* i wybierz *Odwołanie*. Zmień nazwę nowego zapytania na *Schemat*.
5. W pasku formuły powinieneś zobaczyć następującą formułę:
 = Sprzedaż
 Zmień tę formułę w taki sposób:
 = Table.Schema(Sprzedaż)
 Na tym etapie powinieneś zobaczyć schemat tabeli *Sprzedaż*, widoczny na rysunku 11.3.

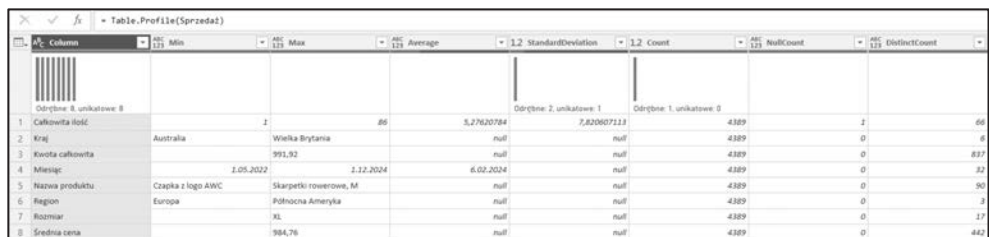


	Name	Position	TypeName	Kind	IsNullable	NumericPrecisionBase	NumericScale
1	Miesiąc	0	Date.Type	date	TRUE		
2	Region	2	Text.Type	text	TRUE		
3	Kraj	2	Text.Type	text	TRUE		
4	Nazwa produktu	3	Text.Type	text	TRUE		
5	Rozmiar	4	Text.Type	text	TRUE		
6	Kwota całkowita	5	Text.Type	text	TRUE		
7	Całkowita ilość	6	Int64.Type	number	TRUE		
8	Średnia cena	7	Text.Type	text	TRUE		

RYСУNEK 11.3. Wynik funkcji *Table.Schema* pokazuje informacje o każdej kolumnie

Zauważ, że kolumna *IsNullable* zawiera tylko jedną wartość: TRUE. Będzie to przydatne w dalszej części ćwiczenia.

6. Kliknij prawym przyciskiem myszy zapytanie *Sprzedaż* i wybierz *Odwołanie*. Zmień nazwę nowego zapytania na *Profil*.
7. W pasku formuły powinieneś zobaczyć następującą formułę:
 = Sprzedaż
 Zmień formułę w taki sposób, by używała funkcji *Table.Profile*:
 = Table.Profile(Sprzedaż)
 Na tym etapie powinieneś zobaczyć różne statystyki kolumn tabeli *Sprzedaż*, widoczne na rysunku 11.4.



	Min	Max	Average	StandardDeviation	Count	NullCount	ErrorCount
1	Całkowita ilość	2	88	5,27620784	7,820807113	4389	0
2	Kraj	Wielka Brytania	null	null	4389	0	0
3	Kwota całkowita	991,82	null	null	4389	0	897
4	Miesiąc	1.01.2022	1.12.2024	6.02.2024	4389	0	33
5	Nazwa produktu	Czapka z logo AKWC	Skarpетки rowersowe, M	null	4389	0	90
6	Region	Europa	Północna Ameryka	null	4389	0	3
7	Rozmiar	XL	null	null	4389	0	17
8	Średnia cena	984,76	null	null	4389	0	442

RYСУNEK 11.4. Wynik funkcji *Table.Profile* pokazuje statystyki dla każdej kolumny

8. W przeciwieństwie do funkcji *Table.Schema* funkcja *Table.Profile* ma opcjonalny drugi parametr, który można wykorzystać do dodania agregatów. Parametr ten akceptuje listę

list, gdzie każda zagnieżdżona lista zawiera nazwę kolumny, selektor typu kolumny oraz funkcję agregującą. Na przykład możemy dodać medianę do naszych statystyk kolumn. Aby to zrobić, zmodyfikuj formułę kroku w następujący sposób:

```
= Table.Profile(Sprzedaż,{{"Median", each true, List.Median}})
```

Ponieważ mediana, podobnie jak wartość minimalna i maksymalna, może być zastosowana do każdej kolumny, nie musimy wybierać żadnych kolumn, więc używamy `each true` jako selektora typu kolumny.

9. Teraz dodajmy liczbę pustych wartości, która różni się od liczby wartości `null`, ponieważ jest stosowana tylko do tekstu. Zmień formułę kroku w następujący sposób (kod poniżej jest sformatowany dla czytelności):

```
= Table.Profile(
    Sprzedaż,
    {
        {"Median", each true, List.Median},
        {"EmptyCount", each Type.Is(_, type nullable text),
        ↪each List.Count(List.Select(_, each = ""))}
    }
)
```

Tym razem zamiast `each true` użyłeś `each Type.Is(_, type nullable text)`, aby wybrać tylko kolumny typu tekstowego. Dlaczego `nullable text`, a nie po prostu `text`? Ponieważ jak wspomniano w kroku 5. tego ćwiczenia, wszystkie kolumny w tabeli *Sprzedaż* mogą przyjmować wartości `null`, więc musieliśmy uwzględnić typ `nullable`, w przeciwnym razie nowa kolumna *EmptyCount* zawierałaby tylko wartości `null`.

10. Zauważ, że w kolumnie *Rozmiar* znajduje się 127 pustych wartości, jak pokazano na rysunku 11.5. Zobacz ćwiczenie 11.1, część 2., aby przypomnieć sobie, jak możesz poradzić sobie z pustymi wartościami.

Column	Min	Max	Average	StandardDeviation	Count	NullCount	DistinctCount	Median	EmptyCount
1. Calkowita ilość	2	88	5,27620794	7,82080713	4389	2	88	2	null
2. Kraj	Australia	Wielka Brytania	null	null	4389	0	2	Australia	0
3. Kwarta całkowita	191,92	112,2024	6,022024	null	4389	0	837	325,99	0
4. Miejsce	1.01.2022	1.12.2024	6.02.2024	null	4389	0	32	1.04.2024	0
5. Nazwa produktu	Czapka z logo AWC	Skarpety rowerowe, M	null	null	4389	0	90	Rower szosowy 250 Czarny, 52	0
6. Region	Europa	Północna Ameryka	null	null	4389	0	3	Pacyfik	0
7. Rozmiar	XL	null	null	null	4389	0	17	52	127
8. Cena	104,76	null	null	null	4389	0	442	325,99	0

RYСУNEK 11.5. Wynik funkcji *Table.Profile* zawiera teraz dodatkowe agregacje



Uwaga. Domyślnie funkcja *Table.Profile* działa na całej tabeli. Jeśli chcesz przeprowadzić profilowanie tylko dla pierwszych 1000 wierszy, musisz je przefiltrować w następujący sposób: *Table.Profile(Table.FirstN(Sprzedaż, 1000))*. Możesz również przed użyciem funkcji *Table.Profile* skorzystać ze znajdującej się na karcie *Narzędzia główne* opcji *Zachowywanie pierwszych wierszy* na liście rozwijanej *Zachowaj wiersze*.

Gotowe rozwiązanie jest zapisane w plikach *R11-03 - rozwiązanie.xlsx* oraz *R11-03 - rozwiązanie.pbix*.

Podsumowanie

W tym rozdziale poznałeś profilowanie danych w Power Query i jak można je wykorzystać przed przystąpieniem do analizy i wizualizacji danych. Jak niektórzy mówią: „śmieci na wejściu, śmieci na wyjściu”. Przed przeprowadzeniem analizy powinieneś zweryfikować swoje dane i sprawdzić, czy nie ma nieoczekiwanych wartości oraz czy jakość danych jest ogólnie zadowalająca.

Chociaż istnieje wiele sposobów profilowania danych — nawet wylistowanie unikatowych wartości w kolumnie jest techniką profilowania danych — edytor Power Query oferuje kilka funkcji, które mogą uczynić to zadanie mniej pracochłonnym. Power Query może pokazać jakość kolumny, która zawiera liczbę wartości prawidłowych, błędów i pustych wartości w kolumnie; rozkład wartości, który pokazuje wykres wartości z ich częstotliwością występowania; oraz statystyki kolumny, które są obliczane dla każdej kolumny i obejmują — choć się do nich nie ograniczają — minimum, maksimum i liczbę pustych wartości.

Jeśli potrzebujesz więcej informacji o danych, a sam interfejs użytkownika nie wystarcza, możesz skorzystać z dwóch funkcji: `Table.Schema` i `Table.Profile`. Pierwsza z nich opisuje kolumny w tabeli, gdzie typ kolumny może być szczególnie przydatny, a druga pokazuje statystyki dla wszystkich kolumn w tabeli, a nawet pozwala Ci określić własne agregacje dla każdej kolumny.

Gdy będziesz zadowolony z jakości swoich danych, możesz zacząć wyciągać z nich wnioski. W rozdziale 12. „Podstawy analizy tekstu”, nauczysz się kilku technik analizy tabel z danymi tekstowymi.

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 

Power Query i Power BI: więcej i lepiej w KRÓTSZYM CZASIE!

Czy chcesz dokonywać cudów z danymi, unikać powtarzalnej pracy i oszczędzać nawet 80 procent czasu? Zautomatyzować kopiowanie i żmudne oczyszczanie danych, aby doprowadzić je do odpowiedniego stanu? A może chcesz ze spokojem podchodzić do podejmowania racjonalnych decyzji opartych na ogromnych ilościach niejednorodnych danych?

Teraz możesz łatwo sprostać takim wyzwaniom i uzyskiwać niezwykle rezultaty. Wystarczy, że skorzystasz z narzędzi dostępnych w Excelu — i z tej książki. To zaktualizowane wydanie praktycznego przewodnika po Power Query. Pozwoli Ci opanować umiejętność efektywnego przetwarzania danych i wydobywania z nich wartościowych informacji. Dowiesz się, jak wykonywać złożone prace analityczne i jak używać procedur w języku M. Znajdziesz tu ponad 70 ćwiczeń i 200 plików, dzięki którym nauczysz się importować bezładne, rozdzielone tabele danych i automatycznie przekształcać je w ustrukturyzowane zbiory gotowe do analizy. Większość opisanych technik z łatwością zapamiętasz i użyjesz do swoich potrzeb.

W książce między innymi:

- solidne podstawy pracy z narzędziem Power Query
- proste i zaawansowane techniki dekompozycji tabel
- tworzenie elastycznych procedur w języku M
- pozyskiwanie informacji z danych tekstowych i sieci społecznościowych
- profilowanie danych i poprawianie

DANIIL MASLYUK jest niezależnym konsultantem do spraw analityki biznesowej, trenerem i mówcą. Napisał wszystkie oficjalne książki przygotowujące do egzaminów Microsoft Power BI. Otrzymał tytuł Microsoft MVP.

GIL RAVIV jest cenionym ekspertem w zakresie danych i sztucznej inteligencji. Otrzymał pięcioletnią nagrodę Microsoft MVP. Jest też właścicielem czterech patentów w dziedzinie sieci komputerowych, cyberbezpieczeństwa i analityki.

Helion	KOD KORZYŚCI Sięgnij po więcej! ▶	
 helion.pl	ISBN 978-83-289-2698-1	
 HELION S.A. ul. Kościuszki 1c 44-100 Gliwice tel.: 32 230 98 63 helion@helion.pl		
Cena: 109,00 zł		

Microsoft Press