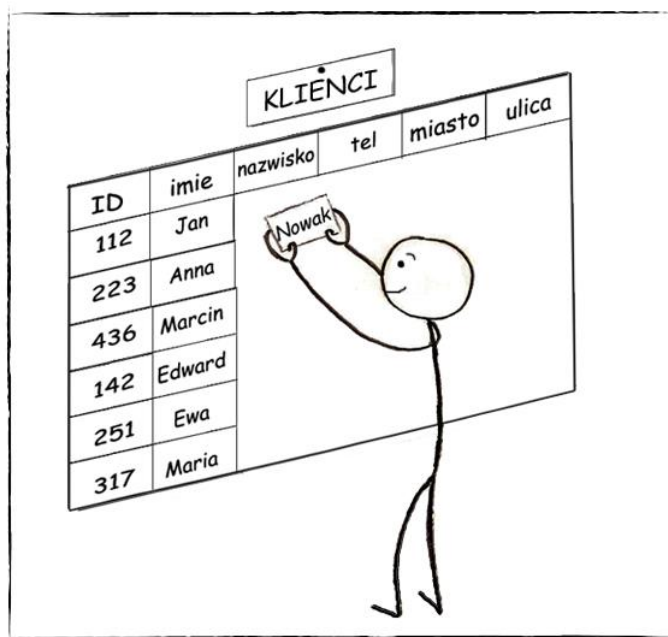


8. Implementujemy tabele

W tym rozdziale znajdziesz odpowiedzi na następujące pytania:

- Czym jest tabela i jakie są jej elementy składowe?
- Jaki jest związek pomiędzy pojęciowym modelem danych, a tabelami w systemie zarządzania relacyjną bazą danych?
- Z jakich kroków składa się proces implementacji tabel?
- Jak tworzyć tabele zapewniające danym integralność dziedzinową oraz encji?
- Jak optymalizować czas dostępu do danych za pomocą indeksów?
- W jaki sposób tworzyć związki pomiędzy tabelami oraz zapewniać integralność referencyjną?
- Jak importować do bazy zbiory danych utworzone w innych aplikacjach?



W niniejszym rozdziale zostanie przybliżone pojęcie tabeli, która jest podstawowym elementem składowym systemów zarządzania relacyjną bazą danych. Dzięki tabelom możemy przechowywać w sposób usystematyzowany zbiory danych, z których korzystają pozostałe obiekty bazy, takie jak: kwerendy, formularze, raporty, makra i moduły. Poznamy zasady tworzenia tabel oraz dobierania odpowiednich typów i określania właściwości pól przechowujących dane w tabelach. Przedstawione zostanie pojęcie klucza podstawowego tabeli, jego cechy oraz sposób definiowania. W dalszej części rozdziału poznamy możliwości tworzenia i wykorzystania związków pomiędzy tabelami. W ramach procesu implementacji tabel, pokazane zostaną również sposoby realizacji mechanizmów odpowiedzialnych za integralność danych. W końcowej części rozdziału zaprezentowano przykład utworzenia tabeli poprzez import danych z pliku tekstowego.

8.1. Anatomia tabel

Tabela jest podstawowym elementem składowym systemu zarządzania relacyjną bazą danych. Tabele wraz z łączącymi je związkami implementują schemat bazy danych, którego kształt decyduje o stopniu spełnienia wymagań informacyjnych organizacji, jakości odwzorowania świata rzeczywistego, którego odbiciem powinna być baza danych. O tabeli możemy myśleć, jako o pojemniku, który zapewnia przechowywanie danych i dostęp do nich w sposób usystematyzowany.



Tabela (ang. *Table*) to zbiór powiązanych ze sobą danych, przedstawiony jako układ poziomych wierszy i pionowych kolumn. Każdy wiersz tabeli zawiera zestaw danych o jednym, konkretnym obiekcie (np. imię, nazwisko, adres, nr telefonu klienta). Każda z kolumn przechowuje wartości opisujące wybraną cechę (atrybut) poszczególnych obiektów (np. imiona wszystkich klientów).

Tabela w systemie zarządzania relacyjną bazą danych jest implementacją relacji (patrz rozdział pt. "Po co nam model danych?").

Jak pamiętamy, w relacyjnych bazach danych, dane są zazwyczaj przechowywane w wielu tabelach. Każda z tabel gromadzi dane na temat zbioru obiektów, takich jak: osoby, przedmioty czy zdarzenia. W ramach każdego ze zbiorów, obiekty te opisane są z wykorzystaniem zestawu takich samych cech (atrybutów). Tabele z kolei są połączone związkami. Warto w tym miejscu przypomnieć, że termin relacyjne bazy danych związany jest z podstawową strukturą modelu danych, jaką jest relacja, a nie jak często się słyszy, w możliwości tworzenia związków łączących tabele bazy danych (w Ms-Access niefortunnie nazwanych relacjami!).

	IdPracownika	Nazwisko	Imie	NazwaOddzialu	DataUrodzenia
	DB03	Bazan	Alicja	DĘBICA	01-wrz-89
	DB04	Czekaj	Bogdan	DĘBICA	08-mar-80
	KO01	Grzesik	Jolanta	KOLBUSZOWA	01-wrz-87
	KO02	Bokwa	Natalia	KOLBUSZOWA	30-maj-86
	KO03	Kowal	Ryszard	KOLBUSZOWA	24-kwi-90
	MI01	Oleksy	Damian	MIELEC	12-maj-88
	MI02	Zagórski	Mikołaj	MIELEC	22-lut-92

Rysunek 68. Tabela w widoku arkusza danych

W strukturze tabeli wyróżnić możemy elementy w układzie pionowym – kolumny i poziomym – wiersze (Rysunek 68). Pierwszy wiersz tabeli nazywany jest nagłówkiem i zawiera nazwy kolumn (pól) tabeli. Pozostałe wiersze tabeli (nazywane rekordami) przechowują wartości cech (atrybutów) kolejnych obiektów (np. identyfikatory, imiona, nazwiska i stanowiska pracowników, daty zdarzeń, transakcje sprzedaży). Często terminy wiersz i rekord używane są zamiennie; w rzeczywistości termin wiersz odnosi się do tabeli, jako struktury, natomiast rekord to zestaw danych zawartych w wierszu. Podobnie jest w przypadku kolumn i pól – pierwszy termin opisuje część struktury tabeli, a drugi określa dane zawarte w kolumnie.

8.2. Proces tworzenia tabel

System zarządzania relacyjną bazą danych Ms-Access umożliwia tworzenie tabel na kilka, przedstawionych poniżej sposobów:

- podczas tworzenia nowej bazy danych,
- przez wprowadzenie przykładowych danych,
- za pomocą szablonów tabel,
- przez import danych,
- w widoku projektu nowej tabeli.

Tworzenie tabeli w widoku projektu daje największe możliwości w zakresie definiowania struktury powstającej tabeli. Można precyzyjnie określać zarówno typy danych pól, jak i ich właściwości, dzięki którym zapewniana jest integralność danych. W związku z tym, przede wszystkim ten sposób implementacji został szczegółowo omówiony w dalszej części rozdziału.

Proces tworzenia tabeli, realizowany w widoku projektu odbywa się w oparciu o następujące kroki:

- 1) Utworzenie nowej tabeli
- 2) Definicja pól tabeli
 - a) Dobór nazw pól
 - b) Dobór typów danych
 - c) Definicja właściwości pól
- 3) Wybór klucza podstawowego tabeli
- 4) Optymalizacja dostępu do danych

Po implementacji struktury tabel, w celu realizacji kompletnego schematu bazy danych, w kolejnym kroku definiowane są związki występujące pomiędzy tabelami. W dalszej części rozdziału szczegółowo scharakteryzowano każdy z wyżej wymienionych kroków.

8.2.1. Utworzenie nowej tabeli

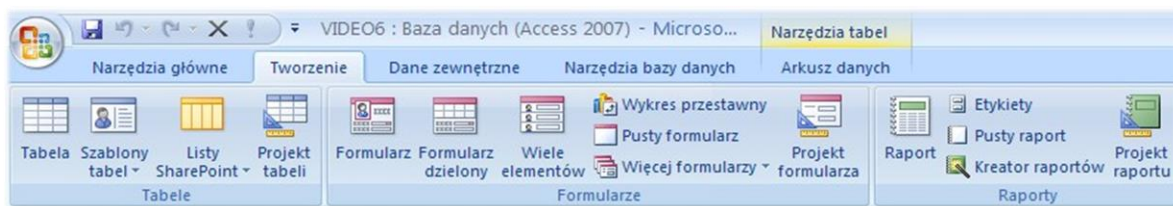
W bazach danych zdecydowanej większości firm można znaleźć następujące tabele:

- **Pracownicy** – tabela zawierająca dane pracowników firmy,
- **Klienci** – tabela przechowująca dane klientów firmy,
- **Produkty** – tabela z danymi towarów oferowanych przez firmę,
- **Zamowienia** – tabela z danymi na temat zamówień składanych przez klientów,
- **Faktury** – tabela będąca rejestrem faktur wystawionych przez firmę.

Jak pokazują powyższe przykłady, nazwy tabel powinny być samoopisujące się – dobrane tak, aby nie było wątpliwości, jakie jest przeznaczenie każdej tabeli. Widać również, że w przedstawionych nazwach nie użyto polskich znaków (tzw. ogonków). Pozwala to uniknąć wielu kłopotów, w przypadku przenoszenia bazy danych do innego systemu zarządzania relacyjną bazą danych.

Nazwy tabel tworzonych w Ms-Access mogą zawierać do 64 znaków. Nazwa tabeli może składać się z liter i cyfr, oraz znaków specjalnych, m.in. takich jak odstęp, podkreślenie, myślnik oraz nawiasy.

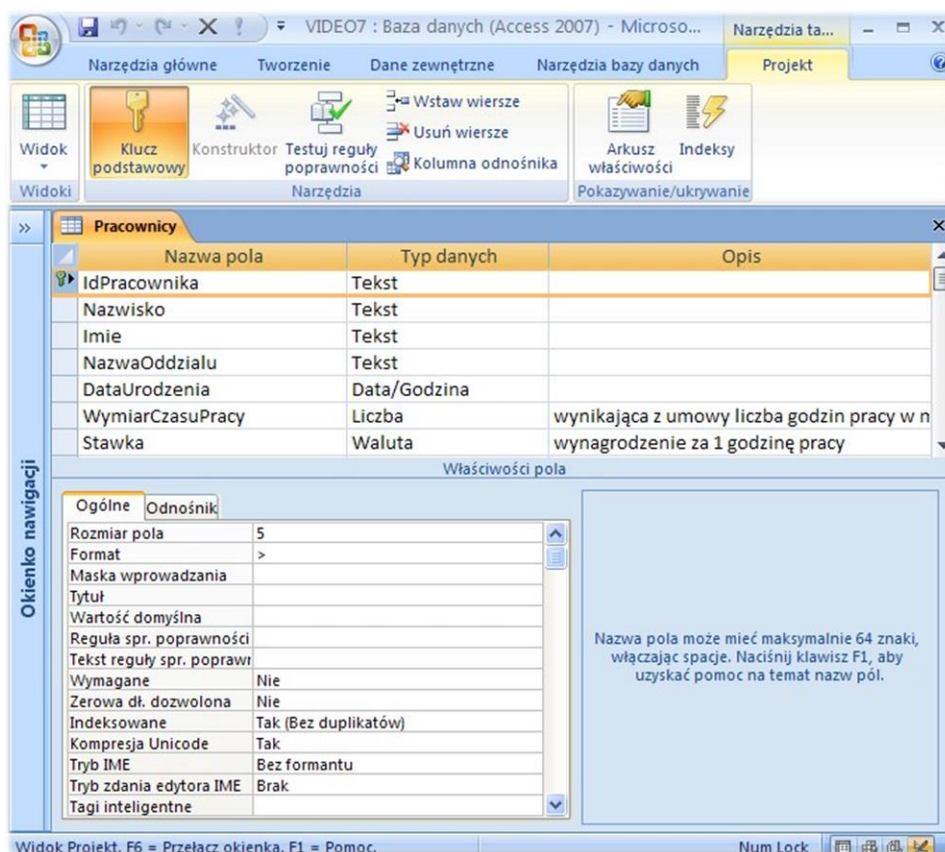
Tworzenie nowej tabeli w widoku projektu inicjujemy wybierając przycisk *Projekt tabeli* na zakładce *Tworzenie*.



Rysunek 69. Wstążka zawierająca opcje Tworzenia tabel

8.2.2. Definicja pól tabeli

Po ustaleniu listy obiektów, których dane będziemy gromadzić, należy określić cechy, które najlepiej je opisują. Przykładowo w przypadku zbierania danych klientów istotna będzie znajomość ich imion, nazwisk, adresów, numerów NIP (te dane są niezbędne do wystawienia faktury). Znaczenie mogą mieć również takie dane jak adres mailowy czy numer telefonu (dające możliwość szybkiego kontaktu z klientem). Przykłady cech nieistotnych (w większości przypadków) to kolor oczu, czy długość włosów klienta. Jak już wcześniej wspomniano (patrz rozdział pt. „Modelujemy dane”), należy pamiętać, że dobór atrybutów jest uwarunkowany kontekstem projektowym i zależy od specyfiki działalności organizacji wykorzystującej dane. Przykładowo dla biura matrymonialnego, serwisu randkowego, czy witryny programu Top Model, cechy uznane powyżej za nieistotne mogą mieć duże znaczenie i powinny być uwzględnione w projekcie tabel.



Rysunek 70. Siatka projektowa tabeli Pracownicy

Pola tabeli definiujemy w siatce projektowej składającej się z kolumn: *Nazwa pola*, *Typ danych* oraz *Opis*. W każdym wierszu siatki zapisuje się definicję jednego pola tabeli. Widok siatki projektowej tabeli przedstawiono na rysunku 70.

8.2.3. Dobór nazw pól

W pierwszej kolumnie siatki projektowej wprowadzamy nazwy pól tworzonej tabeli. Przy określaniu nazw pól obowiązują te same zalecenia i ograniczenia, które przedstawiono w odniesieniu do nazw tabel. Po wprowadzeniu zestawu pól do siatki projektowej (podaniu nazw wszystkich pól) możliwe są operacje korekty nazw, usunięcia błędnie wprowadzonych pól oraz zmiany kolejności pól.

Kolumna **Opis** wykorzystywana jest w przypadkach, gdy nazwy pól są zapisane skrótowo (np. z uwagi na sposób wyświetlania) lub sposób wykorzystania pola wymaga szerszego wyjaśnienia.

8.2.4. Dobór typów danych

Kolejnym krokiem definiowania tabeli jest określenie typów danych zapisywanych w poszczególnych kolumnach. Dobór odpowiednich typów danych zależy od właściwości oraz sposobu wykorzystania danych i jak pokażą poniższe przykłady, może sprawiać trudności początkującym projektantom baz danych. Najczęściej stosowanym typem danych jest tekst (w Ms-Access jest typem domyślnym), pozwalający na zapis danych tekstowych o długości nieprzekraczającej 255 znaków. Przykładowe dane wprowadzane do pola tekstowego mogą wyglądać następująco:

- *Brzeczyszczkiewicz,*
- *NIETRWAŁY,*
- *65.16,*
- *22 czerwca 1971,*
- *141%,*
- *Brak zwrotu w terminie,*
- *13.98E+04,*
- *NULL,*
- *21-05-98.*

Skoro niektóre wpisy wyglądają jak daty lub liczby, łatwo dojść do mylnego wniosku, że nie powinniśmy sobie komplikować zadania i wszystkie dane w tabeli gromadzić w polach tekstowych. Okazuje się jednak, że nie zawsze jest to dobre rozwiązanie. W pewnych sytuacjach należy zastosować inny typ danych, tj. liczbowy, daty/czasu i innych dedykowanych przechowywaniu określonych rodzajów danych. Jako przykład błędnego zastosowania pola tekstowego można podać sytuację, w której ceny towarów są zapisywane jako tekst. W takim przypadku nie będzie można określić sumarycznej wartości zamówienia. Jest to spowodowane faktem, że operacje arytmetyczne, takie jak mnożenie czy dodawanie, nie mogą być realizowane na danych tekstowych.

Z podobnym ograniczeniem będziemy mieli do czynienia w przypadku zapisu daty zakupu w polu tekstowym, ponieważ nie będzie możliwe wyznaczenie terminu płatności, a gdy zapłata jest zrealizowana nieterminowo, naliczenie odsetek karnych.

Istnieją również sytuacje, w których dane składające się z cyfr (np.: kod pocztowy, numer telefonu, domu czy lokalu) nie będą wykorzystywane w obliczeniach. Wówczas należy wybrać tekstowy typ danych.

Poprawne określenie typu danych pól tabeli pozwala na:

- wykonywanie operacji właściwych dla danych określonych typów,
- uniknięcie niektórych błędów wprowadzania danych,
- efektywne przechowywanie danych,
- precyzyjny i szybki proces pobierania danych.

Ms-Access oferuje szeroką gamę typów danych. Poniżej przedstawiono informacje na temat najczęściej używanych typów danych. Zestawienie dostępnych typów wraz z opisem ich zastosowania i maksymalnym, dopuszczalnym rozmiarem danych zamieszczono na końcu rozdziału.

Tabela 10. Najczęściej wykorzystywane typy danych

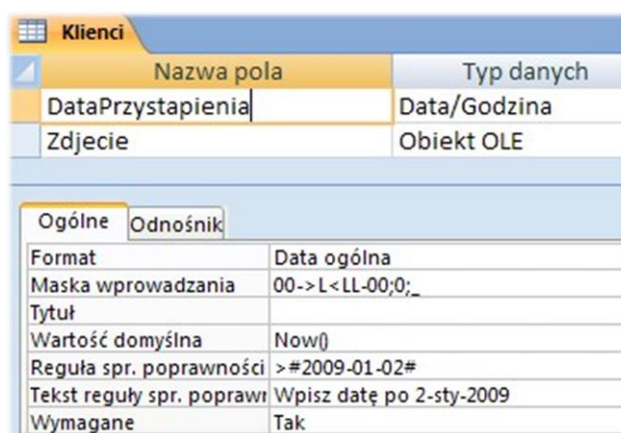
Typ danych	Opis
Tekst	Ciąg dowolnych znaków o długości do 255 znaków
Nota	Długie fragmenty tekstu do 65 535 znaków
Liczba	Dane liczbowe
Walutowy	Dane opisujące wartości pieniężne z oznaczeniem waluty
Data/godzina	Dane opisujące datę i czas
Autonumerowanie	Liczby o unikalnych wartościach generowanych dla każdego kolejnego rekordu (jest zwiększana kolejno o stałą wartość, lub wybierana losowo)
Tak/nie	Jedna z dwóch dostępnych wartości: tak lub nie, prawda lub fałsz

8.2.5. Definiujemy właściwości pól – dbamy o integralność dziedzinową

Po ustaleniu typu przechowywanych w polu danych, należy określić dodatkowe właściwości, definiujące cechy oraz zachowanie pola. Właściwe ustawienie tych parametrów znacznie zwiększa komfort pracy z tabelą oraz w dużym zakresie zabezpiecza przed wprowadzaniem błędnych danych.

Ustawiając właściwości pola określamy rozmiar, sposób wyświetlania danych (format), sposób ich wprowadzania (maska wprowadzania) oraz wartości domyślne. Możemy również kontrolować poprawność wprowadzania danych, przyspieszać wyszukiwanie, czy ustalić, w których polach użytkownik musi wprowadzić określoną wartość. Zdefiniowane właściwości pól mogą być wykorzystane przy projektowaniu formularzy i raportów (mówiąc bardziej precyzyjnie właściwości pól są dziedziczone przez formanty formularzy i raportów, których źródło danych stanowią pola tabeli. Więcej na ten temat w kolejnych rozdziałach).

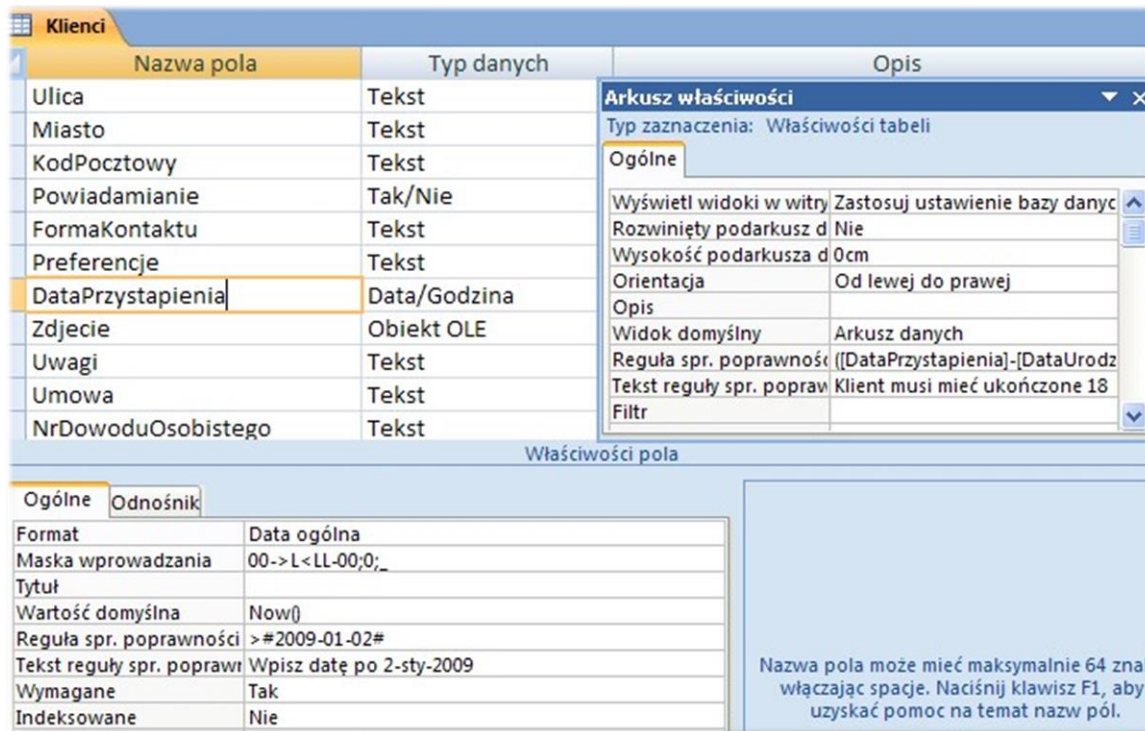
Zestaw właściwości pola jest zależny od typu danych. Rysunek 71. przedstawia właściwości pola typu *Data/Godzina* wraz z przykładowymi wartościami.



Rysunek 71. Określanie właściwości pola

Obszerny opis i przykłady ustawień właściwości najczęściej wykorzystywanych typów pól (tekstowych, liczbowych oraz daty/czasu), zamieszczono na końcu rozdziału.

Dane zgromadzone w tabelach bazy danych muszą spełniać warunki integralności (*ang. integrity constraints*). Ms-Access daje możliwość określenia takich więzów dla pojedynczych pól, całej tabeli oraz związków pomiędzy tabelami.



Rysunek 72. Przykłady więzów integralności dla pojedynczego pola i tabeli bazy danych

Warunki, które muszą spełniać dane wpisywane do pola tabeli określamy ustawiając właściwości pola takie jak: *format*, *maska wprowadzania*, *wymagane* oraz *reguła sprawdzania poprawności*. Przykładowo, jeśli wypożyczalnia filmów została utworzona 2 stycznia 2009 r., data przystąpienia klienta do wypożyczalni musi być równa tej dacie lub następować po niej.

Możemy również zapisać regułę sprawdzania poprawności dla całej tabeli. Reguła taka może weryfikować, czy nie występują sprzeczności między danymi zapisanymi w różnych kolumnach tabeli. Przykładem może być reguła porównywania dat (np. data zwrotu filmu musi nastąpić po dacie jego wypożyczenia, data przystąpienia klienta do wypożyczalni, przy założeniu, że musi on być pełnoletni, powinna następować co najmniej po 18 latach od daty urodzenia). W celu zdefiniowania reguły sprawdzania poprawności należy w widoku projektu tabeli nacisnąć przycisk *Arkusz właściwości*, znajdujący się w grupie *Pokazywanie/ukrywanie*. Po wyświetleniu arkusza właściwości możliwe jest skonstruowanie reguły określającej zależności między danymi w różnych polach tabeli. Poniżej przedstawiono przykładową regułę sprawdzającą, czy klient jest pełnoletni.

$$([DataPrzystapienia] - [DataUrodzenia]) / 365 \geq 18$$

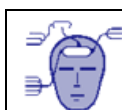
Należy zauważyć, że składnia wyrażeń tworzonych w środowisku Ms-Access jest precyzyjnie określona. Oznacza to, że nazwy pól tabeli użyte w wyrażeniu zawsze powinny być ujęte w nawiasy kwadratowe (zasady te obowiązują również przy definiowaniu kwerend, formularzy i raportów).

Dla więzów integralności określonych dla pojedynczego pola lub więzów tabeli możemy zdefiniować *Tekst reguły sprawdzania poprawności* wyświetlany w formie komunikatu przy próbach wprowadzania przez użytkownika danych naruszających te więzy. Więzy integralności referencyjnej tworzone są w

procesie definiowania związków pomiędzy tabelami i zostaną omówione w dalszej części niniejszego rozdziału.

8.2.6. Definiujemy klucz podstawowy – dbamy o integralność encji

Zgodnie z definicją (Patrz rozdział pt. „Po co nam model danych?”) klucz podstawowy tabeli stanowi pojedyncze pole lub zestaw pól tabeli, który zapewnia jednoznaczność identyfikacji każdego rekordu. Pole kandydujące do pełnienia roli klucza podstawowego musi umożliwiać identyfikację wiersza, nie może zawierać wartości **NULL**, a jego wartość nie powinna być często zmieniana (najlepiej jeśli jest niezmienna). Zadaniem klucza podstawowego jest zapewnienie integralności encji, co jak pamiętamy, w przypadku modelu relacyjnego oznacza, że każda tabela musi posiadać klucz podstawowy, którego wartości muszą być w ramach tabeli niepowtarzalne i różne od **NULL**.

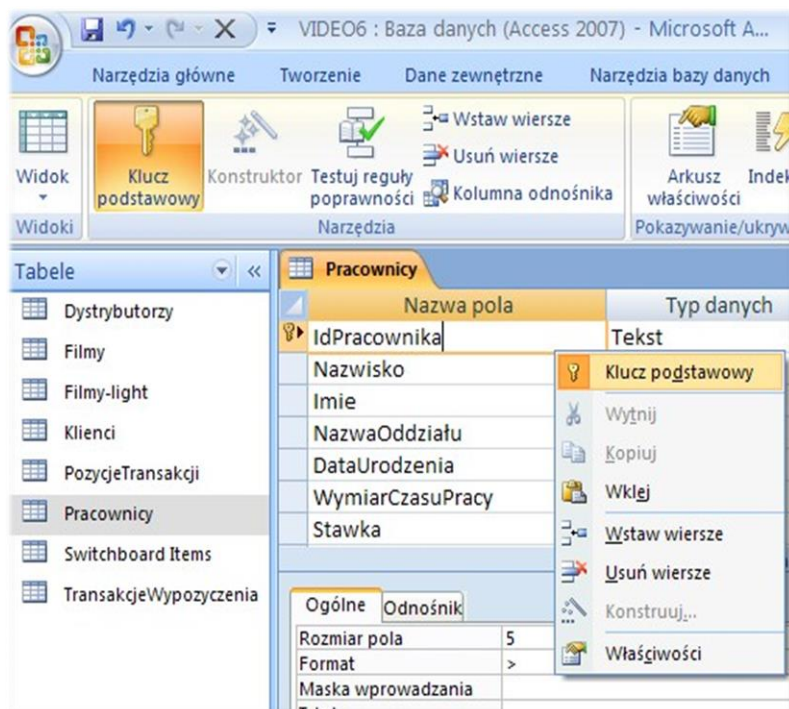


Klucz podstawowy tabeli (ang. *Table's Primary Key*) to co najmniej jedno pole tabeli, którego wartości jednoznacznie identyfikują każdy rekord w tabeli. Wartości klucza podstawowego muszą być różne od **NULL**. Każdy wiersz tabeli musi posiadać niepowtarzalną wartość klucza podstawowego. Każda tabela w relacyjnej bazie danych musi posiadać klucz podstawowy.

Aby wybrać pole lub zestaw pól, które będą pełniły rolę klucza podstawowego należy:

- ❶ Zaznaczyć pole (lub pola) na lewym marginesie siatki projektowej.
- ❷ Przejść na zakładkę *Projekt* i wybrać przycisk *Klucz podstawowy*.

Klucz podstawowy można również ustawić korzystając z menu podręcznego w obszarze wskazanego pola (rysunek 73.).



Rysunek 73. Definiowanie klucza podstawowego

8.2.7. Tworzymy indeksy - przyspieszamy dostęp do danych

Wyszukiwanie danych zapisanych w tabeli można znacznie przyspieszyć przez odpowiednie jej uporządkowanie. Jeżeli przykładowo mamy wybrać dane tych, spośród tysięcy naszych klientów, którzy mieszkają w Rzeszowie, wygodnie będzie to zrobić korzystając z zestawu danych uporządkowanych według pola **Miasto**. W takim przypadku nie jest konieczne przeglądanie całej listy klientów, a jedynie oznaczenie pierwszego wpisu z nazwą szukanego miasta i wszystkich kolejno następujących pozycji, aż do ostatniego wpisu z wartością „Rzeszów”. W ten sposób wybrane zostały dane wszystkich klientów z Rzeszowa, przeglądanie pozostałej zawartości tabeli nie jest konieczne.

Ms-Access przyspiesza operacje wyszukiwania dzięki indeksom, z których korzysta tak, jak czytelnik z indeksu w książce: system odnajduje dane odczytując ich położenie z indeksu, a następnie przechodzi bezpośrednio do lokalizacji poszukiwanego rekordu. Odnalezienie potrzebnych danych przy wykorzystaniu indeksu jest znacznie szybsze niż przez przeszukiwanie wszystkich rekordów tabeli.



Indeks (ang. *Index*) przechowuje informacje o lokalizacji rekordów w tabeli uporządkowane według pola lub pól wybranych do indeksowania.

Ms-Access tworzy automatycznie indeks dla wartości klucza podstawowego, dodatkowe indeksy mogą zostać utworzone przez użytkownika po uwzględnieniu wymagań informacyjnych dla każdej z perspektyw. Poza przyspieszaniem operacji wyszukiwania, indeks powoduje zwykle zmniejszenie wydajności systemu przy modyfikacji danych w tabeli zawierającej pola indeksowane. Wynika to z faktu, że Ms-Access musi uaktualnić indeks przy każdej zmianie, dodaniu, bądź usunięciu rekordu. Z tego powodu indeksy tworzy się w polach, w których przewiduje się przechowywanie wielu różnych wartości, które będą często wyszukiwane i sortowane. Indeksowane mogą być pola typu: Data/godzina, Liczba, Waluta, Autonumerowanie, Tekst, Nota, Hiperłącze oraz Tak/Nie.

Tworzenie *indeksu jednopolewego* realizowane jest przez ustawienie właściwości *Indeksowane* wybranego pola w *Widoku projektu* tabeli.

Właściwości pola	
Ogólne	Oдноśnik
Format	dd\-mmm\)-rr
Maska wprowadzania	
Tytuł	
Wartość domyślna	
Reguła spr. poprawności	
Tekst reguły spr. poprawy	
Wymagane	Nie
Indeksowane	Nie
Tryb IME	Nie
Tryb zdania edytora IME	Tak (Duplikaty OK)
Tagi inteligentne	Tak (Bez duplikatów)
Wyrównanie tekstu	Ogólne
Pokaż okno wyboru daty	Dla dat

Rysunek 74. Ustawianie właściwości pola Indeksowane

Poniżej przedstawiono dostępne ustawienia indeksowania.

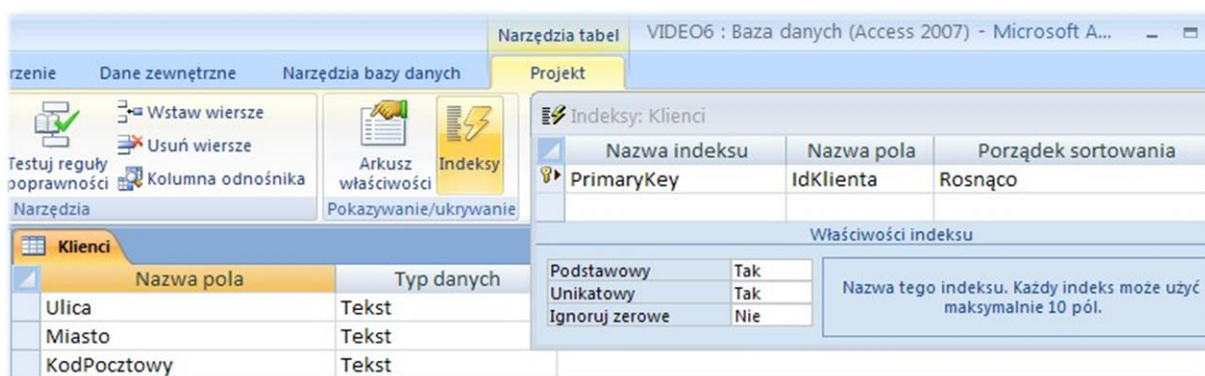
Tabela 11. Ustawienia indeksowania

Ustawienie Indeksowania	Działanie
Tak (Bez duplikatów)	Tworzy unikatowy indeks. Dane w indeksowanym polu muszą być unikalne (nie mogą występować powtórzenia).
Tak (Duplikaty OK)	W polu tworzony jest indeks. Dopuszcza się powtórzenia.
Nie	Indeks nie zostanie utworzony, lub istniejący indeks zostanie usunięty.

Indeks oparty na wielu polach (maksymalnie 10 pól) tworzy się, jeśli przewidywane jest wyszukiwanie, lub sortowanie danych według dwóch lub więcej pól równocześnie.

Aby utworzyć *indeks wielopolowy* (składający się z pól **Miasto** i **Ulica**) w tabeli **Klienci** wykonaj następujące operacje w *Widoku projektu tabeli*:

- 1 Kliknij przycisk *Indeksy* na karcie *Projekt* w grupie *Pokazywanie/Ukrywanie*. Zostanie wyświetlone okno *Indeksy* zawierające listę istniejących indeksów (rysunek 75.).



Rysunek 75. Wyświetlanie listy indeksów tabeli

- 2 W pierwszym wolnym wierszu w kolumnie *Nazwa indeksu* wprowadź nazwę tworzonego indeksu, np. **Adres**.
- 3 W kolumnie *Nazwa pola* dodaj pierwsze pole indeksu, np. **Miasto**
- 4 W kolejnym wierszu zostaw pustą kolumnę *Nazwa indeksu*, a w kolumnie *Nazwa pola* dodaj kolejne pole, np. **Ulica**.

Kolejne pola indeksu wielopolowego dodaje się w analogiczny sposób.

W przypadku definiowania kolejnego indeksu wielopolowego należy podać nazwę indeksu, nazwę indeksowanego pola (lub pól) i określić porządek sortowania (rysunek 76.).



Rysunek 76. Dodawanie indeksu wielopolowego do tabeli

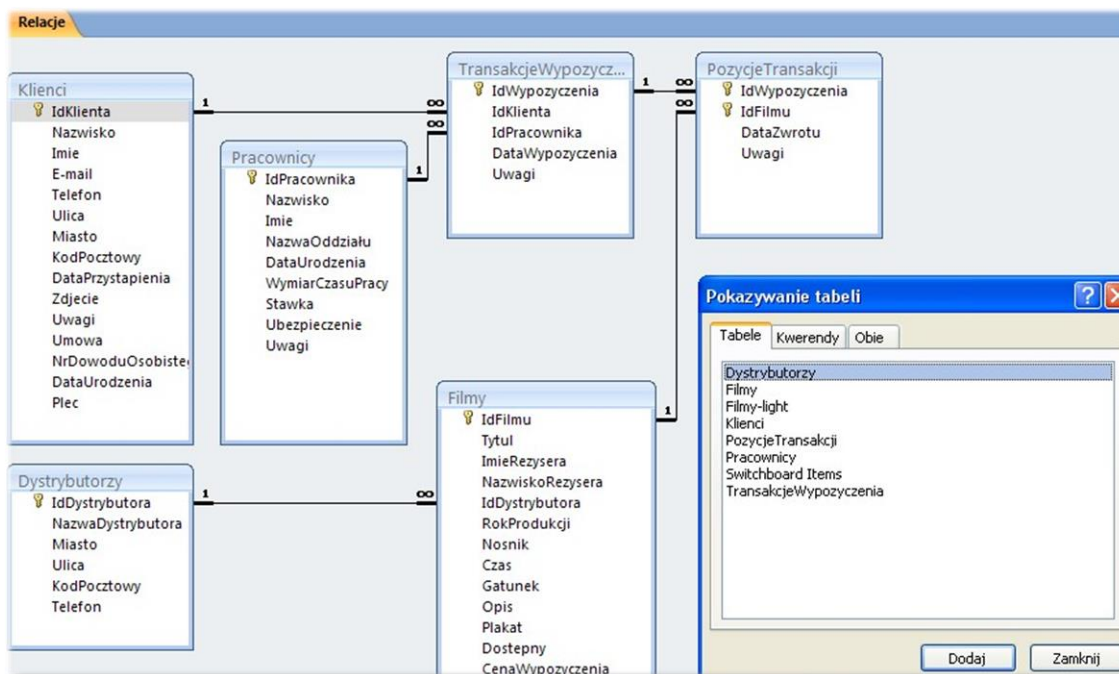
Po ustawieniu wszystkich niezbędnych parametrów związanych z indeksowaniem należy zapisać tabelę i zakończyć pracę z siatką projektową. Zapis inicjowany jest przez naciśnięcie kombinacji *Ctrl+S* lub przycisku z ikoną dyskietki na pasku narzędzi. Podobnie jak w przypadku pozostałych aplikacji Ms-Office, Ms-Access również pyta użytkownika, czy zapisać zmiany przed zamknięciem okna widoku projektu.

8.3. Łączymy tabele

Jak pamiętamy, jedną z cech charakterystycznych modelu relacyjnego jest gromadzenie danych opisujących różne kategorie obiektów w odrębnych tabelach. Dobrze zaprojektowana baza danych powinna stanowić odwzorowanie modelowanej dziedziny przedmiotowej, uwzględniające oprócz samych obiektów również związki zachodzące pomiędzy nimi. Systemy zarządzania relacyjną bazą danych, w tym Ms-Access, umożliwiają implementację związków między tabelami. Związki te odzwierciedlają zależności pomiędzy obiektami, których dane zapisano w postaci rekordów w odpowiednich tabelach. Związki zidentyfikowane na etapie tworzenia modelu danych są implementowane w formie relacji pomiędzy tabelami. Poniższe zestawienie prezentuje przykładowe relacje pomiędzy tabelami relacyjnej bazy danych.

Tabela 12. Przykładowe relacje pomiędzy tabelami

Tabela A	Tabela B	Związek
Dystrybutorzy	Filmy	Dystrybutor <i>dostarcza</i> filmy
Klienci	Filmy	Klienci <i>wypożyczają</i> filmy
Pracownicy	Klienci	Pracownicy <i>rejestrują</i> klientów
Filmy	Transakcje wypożyczenia	Filmy <i>udostępniane</i> w ramach transakcji wypożyczenia



Rysunek 77. Widok związków tabel na karcie relacje

Związki pomiędzy tabelami określa się w Ms-Access na *karcie Relacje* (niefortunna nazwa przyjęta w polskiej wersji systemu Ms-Access i mająca źródło w tłumaczeniu angielskiego słowa *relationships*) w grupie *Narzędzia bazy danych* (rysunek 77.).

Zdefiniowanie związków łączących tabele daje wiele możliwości. Są to między innymi:

- Wyświetlanie powiązanych danych w formularzach i raportach.
- Tworzenie automatycznego warunku złączenia w kwerendach wielotabelowych (korzystających z powiązanych tabel).
- Ustanowienie zarządzanych przez system więzów integralności referencyjnej.

Aby połączyć tabele relacją należy:

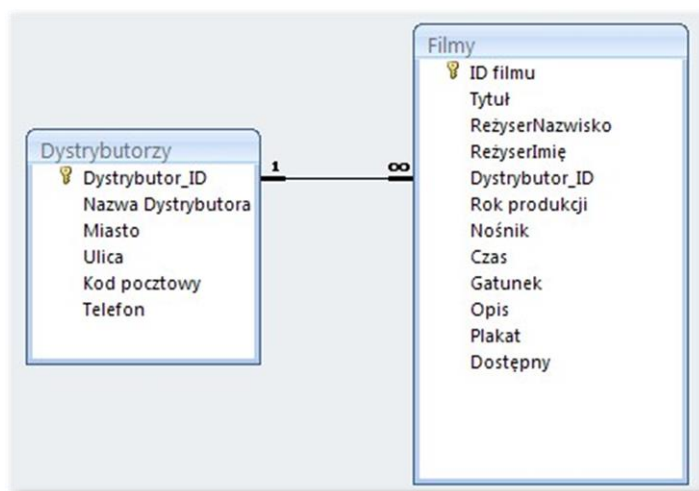
1. Określić zestaw tabel, które chcemy powiązać przez naciśnięcie przycisku *Pokaż tabelę* i wybór z listy dostępnych obiektów tych tabel, które zostaną wykorzystane do utworzenia związków.
2. Przeciągnąć pole klucza podstawowego i upuścić je nad polem klucza obcego. W przypadku omyłkowego wskazania pól tabel, można skorygować błąd wybierając właściwe pole z rozwijanej listy.
3. Zdefiniować w razie potrzeby dodatkowe właściwości związku.

8.3.1. Typy związków pomiędzy tabelami

Istnieją trzy rodzaje związków łączących dwa zbiory obiektów. Związki te wynikają z arności perspektyw związków zdefiniowanych w modelu danych. Są to:

1. Związek *jeden do jednego* (1..1:1..1), w którym jednemu obiektowi ze zbioru A odpowiada jeden i tylko jeden obiekt ze zbioru B (np. każdy produkt posiada gwarancję, każda gwarancja dotyczy jednego produktu).
2. Związek typu *jeden do wielu* (1..1:1..N), w ramach którego z jednym obiektem ze zbioru A jest powiązany jeden lub więcej obiektów ze zbioru B (film na obszarze danego państwa ma jednego dystrybutora, ale jeden dystrybutor może mieć w ofercie wiele filmów).
3. Związek typu *wiele do wielu*, w którym dowolny obiekt ze zbioru A może być powiązany z wieloma obiektami ze zbioru B i na odwrót (każdy pracownik może obsługiwać wielu klientów, każdy z klientów może być obsługiwany przez wielu pracowników).

Jak pamiętamy z poprzedniej sekcji, tworzenie związków w środowisku Ms-Access nie jest skomplikowaną czynnością, a ich zdefiniowanie przynosi wiele korzyści. Sprawna implementacja związków w bazie danych wymaga jednak znajomości kilku pojęć, do których należą: *tabela nadrzędna*, *tabela podrzędna*, *klucz obcy* oraz *zależność*. Pojęcia te zostaną wyjaśnione w oparciu o poniższy przykład związku łączącego tabele **Filmy** oraz **Dystrybutorzy**.



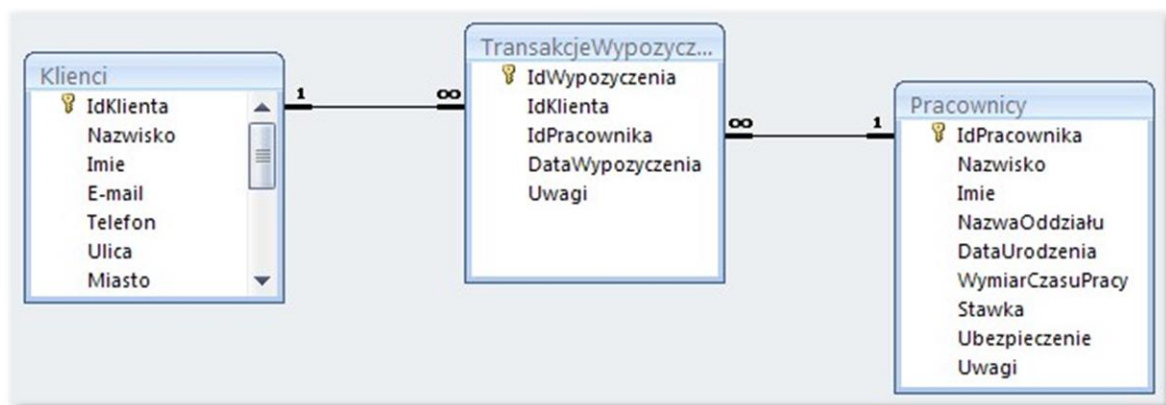
Rysunek 78. Przykład związku jeden do wielu

W przedstawionym związku tabela **Filmy** jest *podrzędna* wobec tabeli **Dystrybutorzy**. Tabela **Dystrybutorzy** jest *nadrzędna* wobec tabeli **Filmy**. W związkach typu jeden do wielu tabela nadrzędna zawsze znajduje się po stronie jeden a tabela podrzędna po stronie wiele. Tabela podrzędna musi posiadać co najmniej jeden *klucz obcy*, czyli pole (lub zbiór pól) o wartościach z tej samej dziedziny, co klucz podstawowy tabeli nadrzędnej. Klucz obcy służy do ustalenia powiązania między tabelami. W przykładzie na rysunku 78. kluczem obcym jest pole **IdDystrybutora** z tabeli **Filmy**. Tworzy ono związek z kluczem głównym tabeli **Dystrybutorzy**.

Powiązane pola muszą spełniać następujące warunki:

- Pole w tabeli nadrzędnej jest kluczem podstawowym lub posiada unikatowy indeks.
- Powiązane pola posiadają zgodne typy danych. Oznacza to, że pola wykorzystane przy łączeniu tabel muszą być tego samego typu. Wyjątkowo w przypadku, gdy w tabeli nadrzędnej pole klucza podstawowego jest typu *Autonumerowanie*, w tabeli podrzędnej pole klucza obcego powinno być typu *Liczba całkowita długa*.

Systemy zarządzania relacyjną bazą danych umożliwiają tworzenie bezpośrednich związków tabel typu *jeden do jednego* oraz *jeden do wielu*, natomiast implementacja związku *wiele do wielu* realizowana jest przez utworzenie dwóch związków typu *jeden do wielu* oraz tabeli implementującej encję asocjacyjną. Poniżej przedstawiono przykład związku pomiędzy klientami i pracownikami wypożyczalni „Omega Video”. Tabela łącząca klientów i pracowników to **TransakcjeWypozyczenia** (Rysunek 79).



Rysunek 79. Realizacja powiązania wiele do wielu (Klienci-Pracownicy)

8.3.2. Zapewniamy integralność referencyjną

W celu zapewnienia zgodności danych zapisanych w powiązanych tabelach – integralności referencyjnej – należy w trakcie definiowania związku pomiędzy tabelami zaznaczyć opcję *Wymuszaj więzy integralności*. Po tej operacji można zatwierdzić utworzony związek naciskając przycisk OK w oknie *Edytowanie relacji*.

Utworzenie więzów integralności referencyjnej przynosi następujące korzyści:

- Do tabeli podrzędnej można wstawić nowy rekord z określoną wartością klucza obcego tylko wtedy, gdy w tabeli nadrzędnej istnieje rekord z tą wartością w powiązonym polu. Przykładowo, gdy do tabeli **TransakcjeWypozyczenia** wpisujemy identyfikator klienta wypożyczającego film, dane klienta z tym identyfikatorem muszą już występować w tabeli **Klienci**.
- Nie można usunąć rekordu z tabeli nadrzędnej, jeśli tabela podrzędna zawiera powiązane rekordy.



Rysunek 80. Określanie właściwości relacji

Dodatkowe właściwości relacji pomiędzy tabelami dotyczą możliwości wykorzystania zdefiniowanych więzów do aktualizacji i usuwania danych w powiązanych tabelach (rysunek 80.).

Ustawienie opcji *Kaskadowo aktualizuj pola pokrewne* powoduje, że przy zmianie wartości klucza głównego rekordu w tabeli nadrzędnej aktualizowane są wartości klucza obcego we wszystkich powiązanych rekordach tabeli podrzędnej. Gdyby na przykład zmieniono identyfikator jednemu z klientów, zmiana ta zostałaby automatycznie wprowadzona we wszystkich danych transakcji wypożyczenia zrealizowanych przez tego klienta.

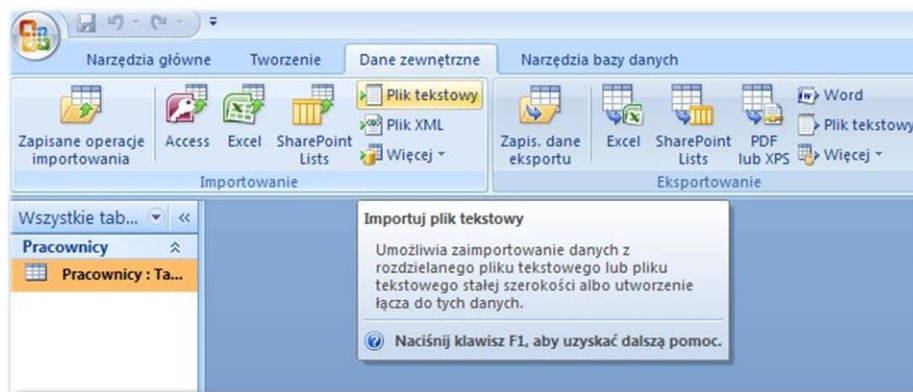
Po zaznaczeniu opcji *Kaskadowo usuń rekordy pokrewne*, podczas usuwania rekordu z tabeli nadrzędnej usuwane są wszystkie powiązane rekordy z tabeli podrzędnej. W związku z możliwością niekontrolowanej utraty danych należy bardzo ostrożnie korzystać z tej opcji.

8.4. Tworzenie tabel przez import danych – idziemy na skróty

Decyzja o rozpoczęciu przygody z bazami danych i wykorzystaniu Ms-Access jest często wynikiem niezadowolenia z dotychczasowych metod przechowywania danych (np. w arkuszach Ms-Excel, listach SharePoint, plikach XML, tekstowych lub tradycyjnej papierowej formie). W takiej sytuacji posiadamy już zbiór danych, który chcemy przenieść do bazy danych programu Ms-Access, w miarę możliwości jak najmniejszym nakładem pracy. Operacje tego typu są wykonywane z wykorzystaniem grupy narzędzi *Importowanie* dostępnej na zakładce *Dane zewnętrzne* (rysunek 81.). Jako przykład prześledźmy proces tworzenia tabeli zawierającej dane pobierane z pliku tekstowego.

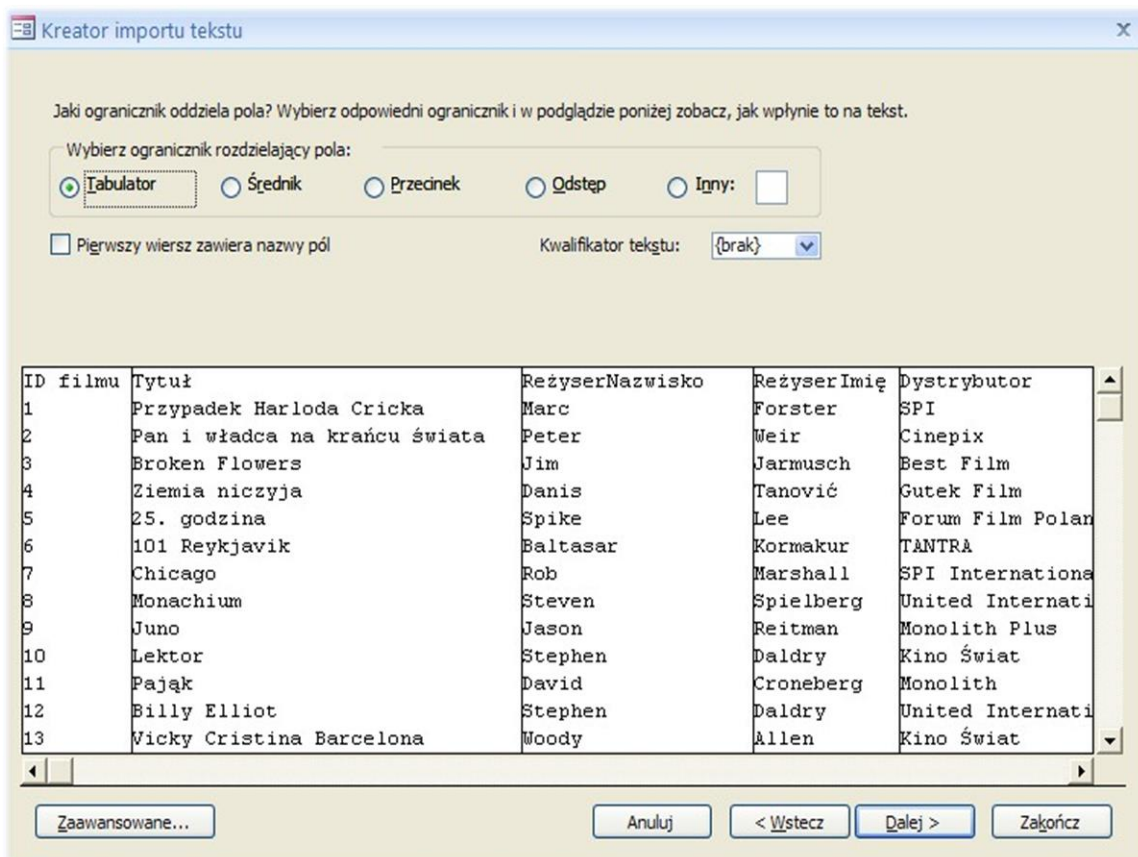
Jest on realizowany w następujących krokach:

- ❶ Uruchom kreatora importu danych przez kliknięcie w przycisk *Plik tekstowy* w grupie *Importowanie* na zakładce *Dane zewnętrzne*.
- ❷ Podaj lokalizację pliku źródłowego i miejsce docelowe danych. Użytkownik powinien w tym miejscu wskazać położenie pliku z danymi i sposób ich wykorzystania – można wypełnić danymi nową tabelę, dołączyć dane do istniejącej tabeli lub utworzyć tabelę zawierającą łącze do źródła danych.
- ❸ Po określeniu tych parametrów wybierz typ separatora (ogranicznika) poszczególnych kolumn z danymi. Możliwe są dwa następujące warianty:
 - Ograniczony – pola są rozdzielane wybranymi znakami np. przecinka lub tabulacji.
 - Stała szerokość – pola są wyrównane w kolumnach.



Rysunek 81. Tworzenie tabeli przez import danych – wybór źródła danych

- 4 W kolejnych krokach pracy kreatora określ znak separatora (może to być tabulator, średnik, przecinek, odstęp lub inny wybrany przez użytkownika znak).
- 5 Zdecyduj czy pierwszy wiersz zestawienia zawiera nazwy pól tabeli.
- 6 Sprecyzuj właściwości kolejnych pól, określając nazwę, typ danych, sposób indeksowania. Możesz również zdecydować o pominięciu pola.
- 7 W końcowych krokach importu danych wybierz klucz podstawowy i podaj nazwę tworzonej tabeli.

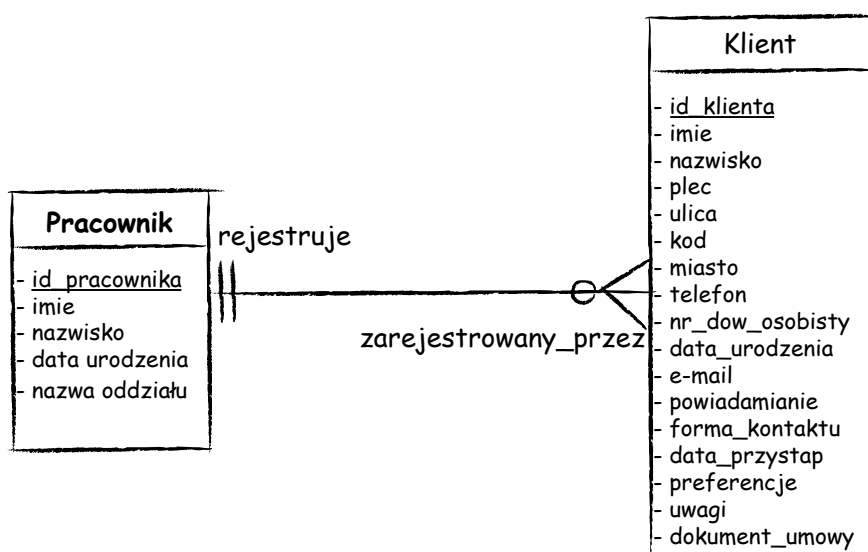


Rysunek 82. Tworzenie tabeli przez import danych – wybór znaku rozdzielającego dane

8.5. Studium przypadku „Omega Video” – implementacja tabel

8.5.1. Implementacja tabel dla perspektywy „Przyjmowanie nowego członka”

W rozdziale pt. „Modelujemy dane” opracowano model danych dla procesu przyjmowania nowego członka wypożyczalni. Modelowanie danych prowadzone w oparciu o analizę wywiadu przeprowadzonego z asystentem klienta oraz wykorzystywane w procesie dokumenty (formularz rejestracyjny, instrukcję wypełniania formularza oraz regulamin wypożyczalni), zostało zwieńczone identyfikacją encji wykorzystywanych w procesie. Analiza procesu przyjmowania nowego członka pozwoliła również zidentyfikować atrybuty encji oraz właściwości związku łączącego encje. Rysunek 83. przedstawia model danych procesu przyjmowania nowego członka.



Rysunek 83. Model danych dla perspektywy „Przyjmowanie nowego członka”

Na diagramie pokazano encje *Klient* i *Pracownik* oraz ich atrybuty. Podkreśleniem wyróżniono atrybuty, które będą pełniły rolę klucza podstawowego. Zaprezentowany diagram związków encji wraz z opisem atrybutów, stanowi solidny fundament do implementacji tabel w środowisku Ms-Access. Zaczynamy od utworzenia tabeli *Klienci*. W oparciu o uwagi dotyczące atrybutów encji pracownik wybrano następujące typy i właściwości danych pól tabeli *Klienci*:


IdKlienta – typ Autonumerowanie. Pole tego typu zapewnia kontrolowane przez system zarządzania bazą danych wstawianie unikalnych wartości. Pola z tym typem danych są godne polecenia dla atrybutów pełniących rolę klucza podstawowego w tabelach z dużą liczbą rekordów, w których zapewnienie unikalności danych przez użytkownika jest utrudnione.

Imie, Nazwisko – pola typu tekstowego, o długości 20 i 25 znaków. Długość pól nie powinna być o wiele większa od najdłuższego znanego nam imienia czy nazwiska. W przypadku napotkania wartości o długości przekraczającej rozmiar pola, rozmiar ten można będzie zwiększyć bez negatywnych konsekwencji dla zgromadzonych w tabeli danych. Stosowanie zbyt długich pól tekstowych może negatywnie wpłynąć na rozmiar pliku bazy danych. Pomimo tego, że część pola będzie niewykorzystana i tak będzie ona miała przydzieloną przez system pamięć. Klienci wypożyczalni będą najczęściej wyszukiwani we-

dług nazwisk, dlatego w polu ustawiono indeksowanie. Wybrano indeksowanie dopuszczające powtórzenia – możemy przecież obsługiwać klientów o tym samym nazwisku. Pole jest wymagane, co oznacza, że dla każdego klienta wartość ta musi być wprowadzona zanim system zapisze rekord w tabeli.


Plec – w tym polu będzie przechowywana informacja o płci klienta. Litera „K” będzie oznaczała, że klient jest kobietą, „M” – mężczyzną. W celu sprawdzania integralności dziedzinowej danych w polu zostanie zapisana reguła poprawności umożliwiająca wprowadzenie wyłącznie liter K lub M o treści: „k” OR „m”. Regułę należy uzupełnić komunikatem wyświetlanym w przypadku wprowadzenia niepoprawnych danych (Tekst reguły sprawdzania poprawności). Pole jest wymagane.

Ulica, Miasto – to pola umożliwiające wprowadzenie składowych adresu klienta. Pola tekstowe o długości 20 znaków, wymagane.



Warto wiedzieć więcej!

Jakie jest najdłuższe imię świata?



Kapitan Fantastyczny Szybszy Od Supermana Spidermana Batmana Wolverine'a Hulka i Flasha razem (wcześniej George Garratt), jest posiadaczem najdłuższego imienia na świecie. Rekord kosztował Kapitana...10 funtów – tyle wynosi koszt brytyjskiej opłaty urzędowej za zmianę imienia. W wersji oryginalnej imię ma 81 liter i brzmi następująco: *Captain Fantastic Faster than Superman Spiderman Batman Wolverine The Hulk and the FLASH COMBINED!* (fot. za telegraph.co.uk).

Kod – Pole tekstowe do przechowywania kodów pocztowych klientów. Długość pola wynosi 6 znaków – tyle potrzeba aby zapisać pięć cyfr kodu oraz znak rozdzielający – myślnik. Ponieważ wszystkie kody pocztowe mają taki sam układ znaków (dwie cyfry, myślnik i trzy kolejne cyfry) w polu należy zastosować maskę wprowadzania, która ułatwi wpisywanie kodów i zabezpieczy przed pomyłkami. Maskę w polu Kod będzie zapisana następująco:

00-000

gdzie „0” oznacza dowolną cyfrę, a „-” myślnik (pełny wykaz symboli wykorzystywanych w masce wprowadzania zamieszczono na końcu rozdziału).

Telefon – tekstowe pole o długości 11 znaków umożliwiające wprowadzenie numeru telefonu klienta. Typ tekstowy umożliwia wprowadzenie do zapisu znaków rozdzielających, zwiększających czytelność zapisu, jak np. myślniki. W polu ustawiono maskę wprowadzania (000-000-000). Pole nie jest wymagane – klientem może być osoba niekorzystająca z telefonu.

NrDowOsobisty – dowód osobisty jest dokumentem stwierdzającym tożsamość i obywatelstwo osoby. Każdy dowód posiada unikalny identyfikator składający się z trzyliterowej nazwy serii i sześciocyfrowego numeru. Pole tekstowe o długości 9 znaków. Maskę wprowadzania pola wygląda następująco:

>LLL000000

gdzie: „>” powoduje zamianę małych znaków na duże, „L” oznacza dowolną literę, a „0” dowolną cyfrę. Pole jest wymagane.

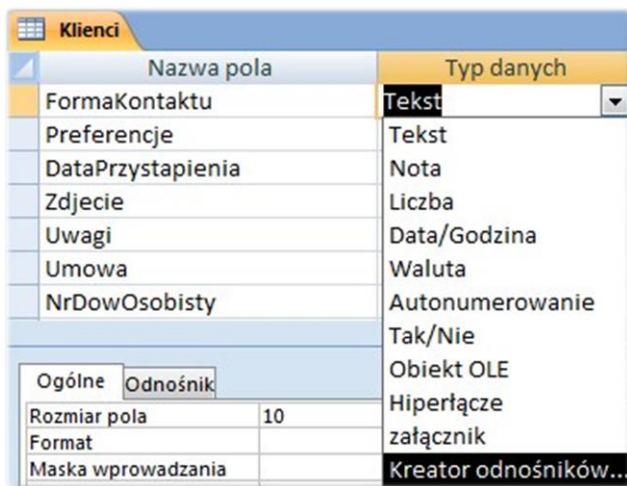
DataUrodzenia – pole typu *Data/Godzina* służące do zapisania daty urodzenia klienta. Pole to razem z polem **DataPrzystapienia** (data przystąpienia klienta do wypożyczalni) umożliwia stworzenie reguły poprawności danych dla tabeli (opartej o dane z różnych pól tabeli). Jeśli zakładamy, że klientami wypożyczalni mogą być wyłącznie osoby pełnoletnie, różnica dat przystąpienia i urodzenia klienta musi być większa lub równa 18 lat. Regułę taką można zapisać następująco:

$([DataPrzystapienia] - [DataUrodzenia]) / 365 \geq 18$

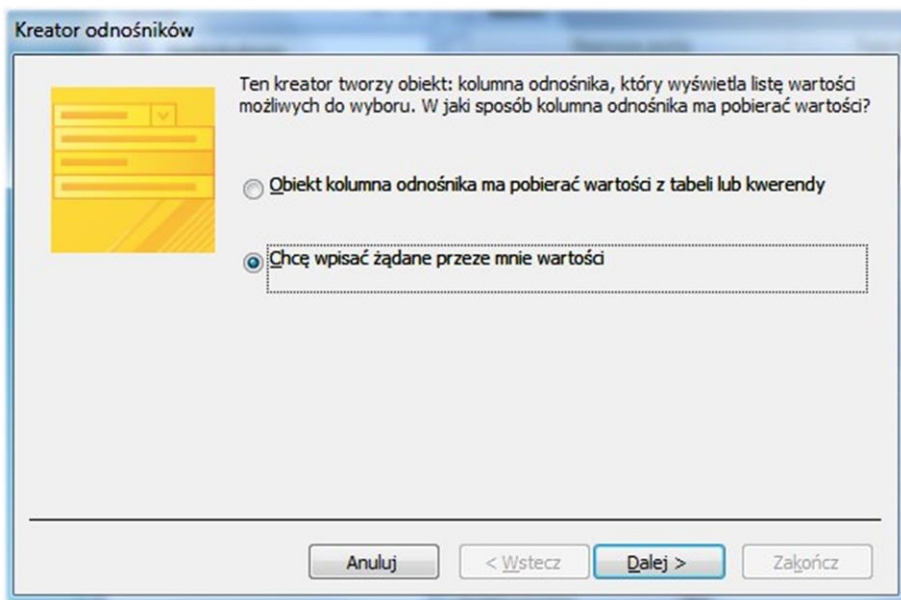
E-mail – pole typu *Hipertącze*. Umożliwia zapisanie adresów mailowych, jak również adresów sieciowych zasobów. Kliknięcie adresu mailowego zapisanego w polu tego typu uruchamia klienta programu pocztowego np. Ms-Outlook lub Mozilla Thunderbird.

Powiadomienie – pole typu *Tak/Nie* pozwalające określić, czy klient ma być informowany w przypadku pojawienia się nowości w interesujących go gatunkach filmowych lub nowych pozycji ulubionego reżysera.

FormaKontakt – w przypadku powiadamiania można w tym polu wpisać preferowaną przez klienta formę kontaktu. Wybieramy typ danych *Tekst*, długość pola 10 znaków. Klient może być powiadamiany telefonicznie, listownie, mailowo lub przez SMS. Skoro pole może zawierać jedną z ustalonych wcześniej wartości, wygodnie byłoby wybierać tę wartość z rozwijanej listy. Taką możliwość daje kreator odnośników, którego znajdziemy na liście typów edytowanego pola.

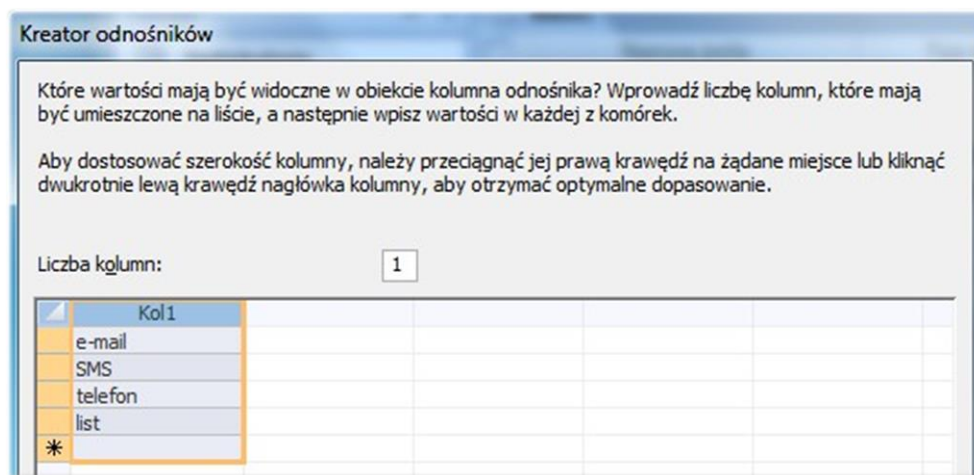


Rysunek 84. Uruchamianie kreatora odnośników



Rysunek 85. Wybór źródła danych dla odnośnika

W pierwszym kroku w oknie dialogowym kreatora zaznaczamy opcję *Chcę wpisać żądane przeze mnie wartości* (rysunek 85.). Taki wybór umożliwia utworzenie listy ustalonych wartości wprowadzanych do pola. Listę taką definiujemy w kolejnym kroku (rysunek 86.). Druga z opcji pojawiających się w oknie kreatora *Obiekt kolumna odnośnika ma pobierać wartości z tabeli lub kwerendy*, pozwala na stworzenie dynamicznej listy wartości pochodzących z pola tabeli lub kwerendy. Wykorzystanie tej opcji zostanie zaprezentowane w dalszej części.



Rysunek 86. Edycja źródła danych odnośnika

W ostatnim kroku pracy kreatora można określić etykietę wyświetlaną z kolumną odnośnika. Domyślnie jest to nazwa pola. W przypadku naszej tabeli pozostawiamy wartość domyślną. Po zakończeniu pracy kreatora pole posiada tekstowy typ danych, a jego długość wynosi 10 znaków. Forma wprowadzania danych ze zdefiniowanej listy wartości (lista rozwijalna) pojawia się w *Widoku Arkusza Danych*.

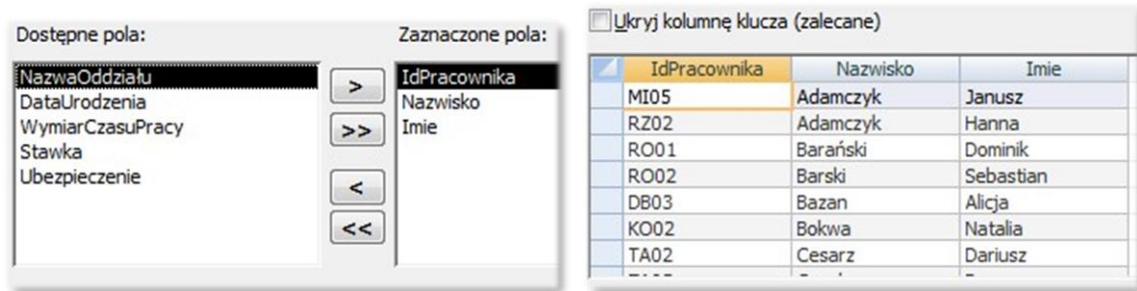
Preferencje – pole tekstowe pozwalające w kilku zdaniach zapisać informacje o preferowanych przez klienta gatunkach filmowych, reżyserach czy aktorach. W polu można zapisać do 255 znaków. Pole nie jest wymagane.

Uwagi – pole typu *Nota*, w którym zapisywane będą uwagi dotyczące korzystania przez klienta z wypożyczalni. Wybrany typ pola umożliwi zapisanie tekstów dłuższych niż 255 znaków. Pole tego typu może pomieścić maksymalnie 65 535 znaków, zapisany tekst może być formatowany. Pole nie jest wymagane.

DokumentUmowy – w tym polu tabeli zapisywany będzie plik zawierający zeskanowaną umowę klienta z wypożyczalnią. Pole typu *OLE* umożliwi wstawienie do rekordu dowolnych obiektów, które są zgodne z mechanizmem OLE.

IdPracownika – pole tekstowe o długości 5 znaków. Pole to będzie pełniło rolę klucza obcego. Jego wprowadzenie jest niezbędne w celu połączenia związkami tabel *Klienci* oraz *Pracownicy*. Jak pamiętamy, aby można było połączyć tabelę pole klucza obcego musi być zgodne dziedziczeniowo z polem klucza podstawowego. Oznacza to, że do pola mogą zostać wprowadzone wyłącznie wartości zapisane uprzednio w polu *IdPracownika* tabeli *Pracownicy*.

Aby ułatwić wprowadzanie danych i użytkownik nie musiał pamiętać wartości wprowadzonych w tabeli *Pracownicy* wykorzystana zostanie lista wartości. W celu zbudowania takiej listy, zostanie użyty kreator odnośników (rysunek 87.). W pierwszym kroku należy wybrać opcję *Obiekt kolumna odnośnika ma pobierać wartości z tabeli lub kwerendy*, a następnie wskazać na źródło danych kolumny odnośnika. Wybieramy pola *IdPracownika*, *Imie* i *Nazwisko* z tabeli *Pracownicy*.



Rysunek 87. Wybór źródła danych oraz podgląd danych kolumny odnośnika

W kolejnym kroku można ustawić sortowanie rekordów. Wybieramy sortowanie wg nazwisk i przechodzimy do podglądu zbioru danych, który będzie wykorzystany do wprowadzania identyfikatorów pracowników do pola `IdPracownika` w tabeli `Klienci`. Kolejnym, bardzo ważnym etapem pracy kreatora jest wybór z przedstawionego zestawienia kolumny zawierającej wartości, które mają być przechowywane w polu `IdPracownika` tabeli `Klienci`. W naszym przypadku jest to pole `IdPracownika`, wartości z pozostałych kolumn będą pełniły funkcję informacyjną (użytkownik wprowadzający identyfikator pracownika będzie widział również jego imię i nazwisko, co znacznie ułatwia wprowadzenia identyfikatora właściwej osoby). Po zatwierdzeniu etykiety przypisanej do pola kończymy pracę kreatora. Pole `IdPracownika` jest wymagane.

Klienci-	
Nazwa pola	Typ danych
IdKlienta	Autonumerowanie
Imie	Tekst
Nazwisko	Tekst
Plec	Tekst
Ulica	Hipertącze
Kod	Tekst
Miasto	Tekst
Telefon	Tekst
NrDowOsobisty	Tekst
DataUrodzenia	Data/Godzina
E-mail	Hipertącze
Powiadamianie	Tak/Nie
FormaKontakt	Tekst
DataPrzystap	Data/Godzina
Preferencje	Tekst
Uwagi	Nota
DokumentUmowy	Obiekt OLE
IdPracownika	Tekst

Rysunek 88. Widok siatki projektowej tabeli `Klienci`

Jak pamiętamy tabela `Pracownicy` posiada strukturę ustaloną przez centralę firmy Omega Video i zawiera ona następujące pola:

- `IdPracownika` – pole tekstowe o długości 10 znaków, wymagane.
- `Imie` – pole tekstowe o długości 15 znaków, wymagane.
- `Nazwisko` – pole tekstowe o długości 15 znaków, wymagane.
- `DataUrodzenia` – pole typu Data/Godzina, wymagane.

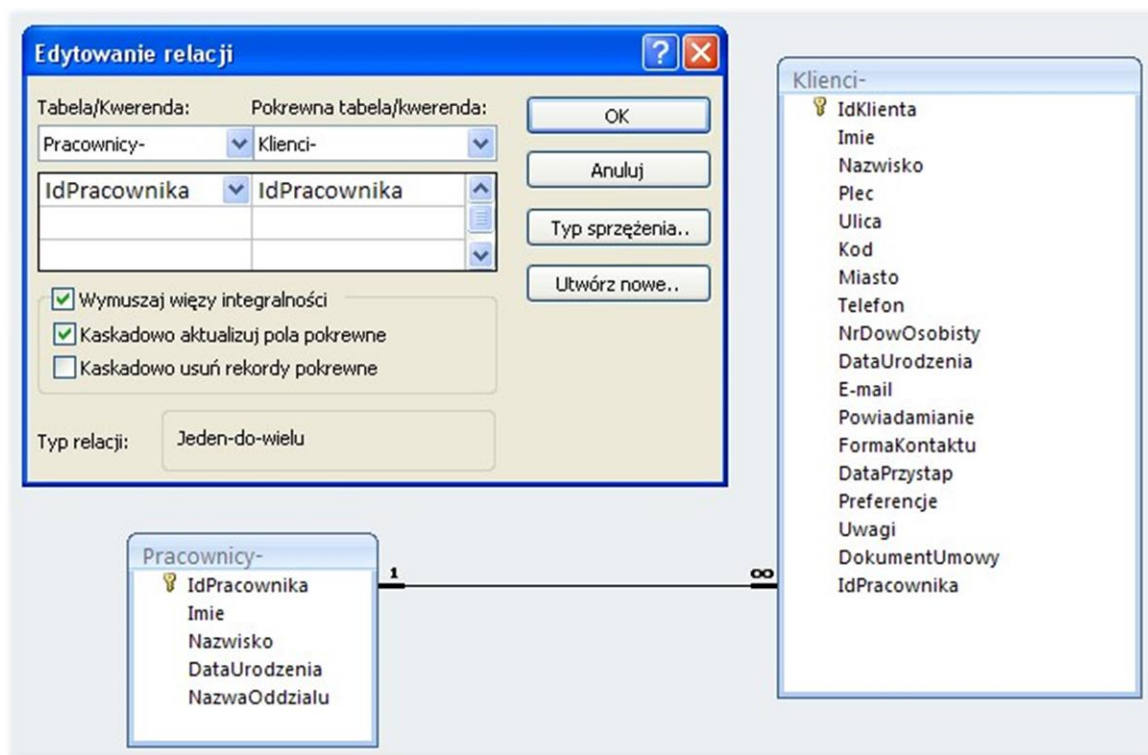
- **NazwaOddzialu** – pole tekstowe o długości 10 znaków, wymagane.

8.5.2. Łączenie tabel związkami

Po utworzeniu tabel możemy przejść do implementacji związku pomiędzy tabelami **Klienci** i **Pracownicy**. Związek przedstawiony na diagramie znajdującym się na początku studium przypadku odpowiada na pytanie: *Których klientów zarejestrował dany pracownik?* Zgodnie z notacją Martina, zapisana w diagramie arność perspektyw związku jest następująca:

- Pracownik może zarejestrować jednego lub więcej klientów. Związek jest opcjonalny, co oznacza, że są pracownicy, którzy nie rejestrują klientów.
- Każdy klient jest rejestrowany przez jednego i tylko jednego pracownika.

Utworzenie takiego związku w programie Ms-Access wymaga dodania pola klucza obcego w tabeli po stronie wiele, czyli w tabeli **Klienci**. Kluczem obcym jest w tym przypadku pole **IdPracownika**. Widok połączonych tabel na karcie *Relacje* oraz okno dialogowe *Edytowanie relacji* zaprezentowano na rysunku 89.



Rysunek 89. Związek tabel **Pracownicy** – **Klienci** zaimplementowany w Ms-Access

Zaznaczenie opcji *Wymuszaj więzy integralności* powoduje sprawdzanie przez system zarządzania bazą danych reguł integralności referencyjnej danych zapisanych w obu tabelach, w polach **IdPracownika**. Aby wspomniane reguły były spełnione, zbiór wartości występujących w polu klucza obcego (w tabeli podrzędnej) musi być podzbiorem zbioru wartości występujących w polu klucza podstawowego (w tabeli nadrzędnej). W przeciwnym razie związek z ustawioną opcją *Wymuszaj więzy integralności* nie zostanie utworzony. Zaznaczenie tej opcji powoduje również wyświetlenie na diagramie połączone tabele informacji o arności związku: symbol „1” po stronie tabeli **Pracownicy** i symbol „∞” po stronie tabeli **Klienci** (rysunek 89.).

Wybór opcji *Kaskadowo aktualizuj pola pokrewne* spowoduje aktualizację wartości powiązanych pól po stronie wiele w przypadku modyfikacji wartości pola po stronie jeden. Z taką sytuacją mielibyśmy do czynienia w przypadku zmiany identyfikatora pracownika. Identyfikator taki mógłby być zmieniany na życzenie pracownika (gdyby np. pracownik z identyfikatorem 13 stwierdził, że nadany numer przynosi mu pecha) lub podczas przeniesienia pracownika do innego oddziału firmy. Opcja ta powoduje utrzymanie przez system spójności danych i w związku z tym została ustawiona.

Pamiętamy, że opcja *Kaskadowo usuń rekordy pokrewne* może wprowadzić w tabelach nieodwracalne zmiany. Jej wybór w przypadku usuwania rekordu z tabeli nadrzędnej (w przypadku firmy Omega Video może tak się zdarzyć podczas zwalniania pracownika) spowoduje usunięcie wszystkich powiązanych rekordów tabeli podrzędnej. W takiej sytuacji tracilibyśmy dane wszystkich klientów przyjętych przez odchodzącego z pracy pracownika. Jest to niedopuszczalne i dlatego też opcja nie została wybrana.

8.6. Pojęcia które warto znać

- Tabela
(ang. *Table*)
- Pole tabeli
(ang. *Table Field*)
- Kolumna tabeli
(ang. *Table Column*)
- Klucz podstawowy tabeli
(ang. *Table primary key*)
- Indeks
(ang. *Index*)
- Indeks jedнопольowy
(ang. *Single-field index*)
- Indeks wielopольowy
(ang. *Multiple-field index*)
- Typ danych
(ang. *Datatype*)
- Maska wprowadzania
(ang. *Input mask*)
- Reguła sprawdzania poprawności
(ang. *Validation Rule*)
- Wiersz Tabeli
(ang. *Table Row*)
- Rekord
(ang. *Record*)
- Tabela podrzędna
(ang. *Child Table*)
- Tabela nadrzędna
(ang. *Parent Table*)
- Nagłówek tabeli
(ang. *Table header*)
- Kolumna odnośnika
(ang. *Lookup field*)
- Więzy integralności
(ang. *Integrity constraints*)
- Format danych
(ang. *Data format*)
- OLE – technologia osadzania obiektów
(ang. *Object Linking and Embedding*)

8.7. Pytania do dyskusji

1. Wyjaśnij w jakim celu wykorzystujemy tabele. Podaj kilka przykładów.
2. Wyjaśnij jak prawidłowo określić zestaw pól tabeli i jaki element specyfikacji systemu jest w tym procesie szczególnie przydatny?
3. Wyjaśnij dlaczego stosujemy różne typy danych? Wymień podstawowe typy danych stosowane w środowisku Ms-Access i omów ich przeznaczenie.
4. Podaj kilka przykładów pól, dla których odpowiedni będzie typ *Liczbowy*.
5. Podaj przykłady zastosowań pól typu *Data/Godzina*.
6. Wyjaśnij jakie warunki powinien spełniać klucz podstawowy tabeli. Podaj kilka przykładów pól, które mogą odgrywać rolę klucza podstawowego tabeli.
7. Wyjaśnij jaką rolę pełni klucz obcy?
8. Wyjaśnij jaka jest różnica pomiędzy wierszem tabeli a rekordem?
9. Wyjaśnij co odróżnia pole od kolumny?
10. Wyjaśnij w jaki sposób indeks ułatwia wyszukiwanie danych? Czy stosowanie indeksu zawsze powoduje zwiększenie wydajności bazy danych? Uzasadnij odpowiedź.
11. Podaj przykłady reguły sprawdzania poprawności w pojedynczej kolumnie i tabeli.
12. Podaj przykłady reguły sprawdzania poprawności dla tabeli.
13. Jakie związki mogą łączyć tabele bazy danych? Podaj przykłady.
14. Jakie jest działanie właściwości związku „Wymuszaj więzy integralności”?
15. Podaj przykład zachowania systemu po wyborze opcji *Kaskadowo aktualizuj pola pokrewne*.

16. Podaj przykład zachowania systemu po wyborze opcji *Kaskadowo usuń rekordy pokrewne*.

8.8. Zadania

1. Korzystając z wybranego przez siebie szablonu wygeneruj system z bazą danych. Przeanalizuj utworzone obiekty bazy danych oraz logikę działania aplikacji. Zaproponuj zmiany w ramach funkcjonalności, które nie są zgodne z Twoimi oczekiwaniami.
2. Utwórz pusty plik bazy danych o nazwie **Omega_Video_1**.
3. Korzystając z bazy danych **Omega_Video_1** utwórz tabelę **Pracownicy** zawierającą następujące pola: **IdPracownika**, **Nazwisko**, **Imie**, **NazwaOddzialu**, **DataUrodzenia**, **WymiarCzasuPracy**, **Stawka**, **Ubezpieczenie**, **Uwagi**. Ustaw odpowiednie typy oraz właściwości pól i zapisz projekt tabeli. Uzupełnij projekt tabeli o dobrane przez siebie pola danych kontaktowych i adresowych oraz nr konta bankowego. Uzasadnij krótko wybrane typy danych. Wpisz przykładowe dane pięciu pracowników.
4. Utwórz tabelę **Wyplaty** zawierającą następujące pola: **IdWyplaty**, **Data**, **Miesiac**, **Kwota**, **IdPracownika**. Ustaw odpowiednie typy i własności pól, zapisz projekt tabeli. Zastanów się, w którym polu można wykorzystać mechanizm odnośnika w celu ułatwienia wprowadzania danych do połączonych tabel.
5. Utwórz związek pomiędzy tabelami **Pracownicy**– **Wyplaty** z wymuszaniem więzów integralności. Wprowadź dane przykładowych wypłat za miesiące od października do grudnia 2011 roku dla dwóch pracowników.
6. Utwórz tabelę **Filmy** przez import danych z pliku **filmy.txt**.
7. Utwórz tabelę **Dystrybutorzy** zawierającą pola: **IdDystrybutora**, **NazwaDystrybutora**, **Miasto**, **Ulica**, **Kod**, **Telefon**. Ustal odpowiednie typy i właściwości pól. Przeprowadź import danych z pliku **dystrybutor.txt** do utworzonej tabeli.
8. Utwórz związek tabel **Filmy** – **Dystrybutorzy** z wymuszaniem więzów integralności. Dobierz odpowiednią arność związku.
9. Dyrektor oddziału firmy Omega Video zdecydował się na rozszerzenie zakresu funkcjonalnego o obszar *Zamawianie pozycji filmowych u dystrybutorów*. W oparciu o diagram związków encji wykonany w ramach zadań rozdziału 6 pt. „*Modelujemy Dane*”, zaimplementuj tabele i związki niezbędne do realizacji tej funkcjonalności. Skorzystaj z tabel utworzonych w poprzednich zadaniach.
10. Korzystając z wywiadu z pomysłodawcą wypożyczalni rowerów RoweRes, oraz globalnego modelu danych systemu opracowanego w rozdziale 6 pt. „*Modelujemy dane*”, utwórz tabele oraz łączące je związki dla dwóch wybranych obszarów funkcjonalnych.
11. W oparciu o diagramy związków encji opracowane w rozdziale *Modelujemy dane* przygotuj schematy bazy danych (tabele i łączące je związki) dla dziedzin przedmiotowych:
 - a. Rejestracja ślubów w urzędzie stanu cywilnego.
 - b. Rejestracja noworodków.
 - c. Przepisy kulinarne (skorzystaj z witryny www.przepisy.pl).
 - d. Prognozę pogody (temperatura, zachmurzenie, opady, kierunek i siła wiatru).
 - e. Zarządzanie planem zajęć na uczelni wyższej.

Tabela 13. Typy danych w Ms-Access

Typ danych	Przechowywana zawartość	Rozmiar
Tekst	Znaki alfanumeryczne. Typ stosowany w przypadku tekstu lub tekstu i liczb, które nie będą używane w obliczeniach (<i>BMW 324, Porsche 911</i>)	Maksymalnie 255 znaków
Nota	Tekst dłuższy niż 255 znaków lub formatowany. Użycie pola Nota jest uzasadnione w przypadku długich opisów i akapitów z formatowaniem, takim jak pogrubienie czy kursywa	65 535, jeśli dane są wprowadzane przez interfejs użytkownika; 2 gigabajty, jeśli dane są wprowadzane automatycznie
Liczba	Wartości liczbowe. Stosowany do danych, które będą używane w obliczeniach	1, 2, 4 lub 8 bajtów (16 bajtów w przypadku identyfikatora replikacji)
Waluta	Wartości pieniężne (waluty)	8 bajtów
Data/Godzina	Każda przechowywana wartość zawiera datę i czas	8 bajtów
Autonumerowanie	Unikatowa wartość liczbową, która jest wstawiana automatycznie, podczas dodawania rekordu. Wartości pola Autonumerowanie mogą być zwiększane kolejno o stałą wartość lub losowo	4 bajty lub 16 bajtów w przypadku identyfikatora replikacji
Tak/Nie	Stosowane w przypadku pól, które mogą zawierać jedną z dwóch możliwych wartości: Tak/Nie, Prawda/Falsz	1 bit
Obiekt OLE	Obiekty OLE (np. wykresy, rysunki, diagramy, arkusze danych) z innych aplikacji systemu Windows	Do 1 gigabajta
Załącznik	Obrazy cyfrowe i inne pliki binarne. Umożliwia przechowywanie wielu załączników dla każdego rekordu.	W przypadku załączników skompresowanych do 2 gigabajtów. W przypadku nieskompresowanych załączników około 700 kilobajtów
Hiperłącze	Hiperłącza pozwalające na dostęp do stron sieci lub do plików. Może być zastosowany do łączenia z obiektami programu Access przechowywanymi w bazie danych	Do 1 gigabajta znaków
Kreator odnośników	Uruchamia Kreatora odnośników, który pozwala utworzyć pole kombi do pobrania wartości z innej tabeli, kwerendy lub listy wartości	Zależnie od tabeli lub kwerendy: rozmiar kolumny związanej

Tabela 14. Właściwości pól tekstowych (innych jeśli określono)

Właściwość	Opis																		
Rozmiar pola	Wartość od 1 do 255. Należy określać dopasowaną (najmniejszą wystarczającą) długość pól tekstowych																		
Rozmiar pola (Liczba)	Określa zakres wprowadzanych danych Bajt 0 do 255 (liczby całkowite) Liczba całkowita -32 768 do 32 767 Liczba całkowita długa -2 147 483 648 do 2 147 483 647 Pojedyncza precyzja $-3,4 \times 10^{38}$ do $3,4 \times 10^{38}$ Podwójna precyzja $-1,797 \times 10^{308}$ do $1,797 \times 10^{308}$ Dziesiętne $-9,999... \times 10^{27}$ do $+9,999... \times 10^{27}$																		
Format	Określa wygląd podczas druku i wyświetlania danych. Można tworzyć własne reguły formatowania z wykorzystaniem symboli: <table border="1" data-bbox="470 667 1385 898"> <thead> <tr> <th>Symbol</th> <th>Opis</th> </tr> </thead> <tbody> <tr> <td>@</td> <td>Wymagane wprowadzenie znaku (litery lub cyfry)</td> </tr> <tr> <td>&</td> <td>Wprowadzenie znaku nie jest wymagane</td> </tr> <tr> <td>></td> <td>Zamiana wprowadzonych znaków na duże</td> </tr> <tr> <td><</td> <td>Zamiana wprowadzonych znaków na małe</td> </tr> </tbody> </table>	Symbol	Opis	@	Wymagane wprowadzenie znaku (litery lub cyfry)	&	Wprowadzenie znaku nie jest wymagane	>	Zamiana wprowadzonych znaków na duże	<	Zamiana wprowadzonych znaków na małe								
Symbol	Opis																		
@	Wymagane wprowadzenie znaku (litery lub cyfry)																		
&	Wprowadzenie znaku nie jest wymagane																		
>	Zamiana wprowadzonych znaków na duże																		
<	Zamiana wprowadzonych znaków na małe																		
Format (Data/godzina)	Dla pól typu data godzina dostępne są predefiniowane formaty: <table border="1" data-bbox="470 947 1005 1189"> <tbody> <tr> <td>Data ogólna</td> <td>2011-03-13 18:15:22</td> </tr> <tr> <td>Data długa</td> <td>13 marca 2011</td> </tr> <tr> <td>Data średnia</td> <td>13-mar-11</td> </tr> <tr> <td>Data krótka</td> <td>2011-03-13</td> </tr> <tr> <td>Godzina długa</td> <td>18:15:22</td> </tr> <tr> <td>Godzina średnia</td> <td>06:15</td> </tr> <tr> <td>Godzina krótka</td> <td>18:15</td> </tr> </tbody> </table>	Data ogólna	2011-03-13 18:15:22	Data długa	13 marca 2011	Data średnia	13-mar-11	Data krótka	2011-03-13	Godzina długa	18:15:22	Godzina średnia	06:15	Godzina krótka	18:15				
Data ogólna	2011-03-13 18:15:22																		
Data długa	13 marca 2011																		
Data średnia	13-mar-11																		
Data krótka	2011-03-13																		
Godzina długa	18:15:22																		
Godzina średnia	06:15																		
Godzina krótka	18:15																		
Format (Liczba)	<table border="1" data-bbox="470 1200 877 1442"> <tbody> <tr> <td>Liczba ogólna</td> <td>3456,789</td> </tr> <tr> <td>Walutowy</td> <td>3 456,79 zł</td> </tr> <tr> <td>Euro</td> <td>3 456,79 €</td> </tr> <tr> <td>Stałoprzecinkowy</td> <td>3456,79</td> </tr> <tr> <td>Standardowy</td> <td>3 456,79</td> </tr> <tr> <td>Procentowy</td> <td>123,00%</td> </tr> <tr> <td>Wykładniczy</td> <td>3,46E+03</td> </tr> </tbody> </table>	Liczba ogólna	3456,789	Walutowy	3 456,79 zł	Euro	3 456,79 €	Stałoprzecinkowy	3456,79	Standardowy	3 456,79	Procentowy	123,00%	Wykładniczy	3,46E+03				
Liczba ogólna	3456,789																		
Walutowy	3 456,79 zł																		
Euro	3 456,79 €																		
Stałoprzecinkowy	3456,79																		
Standardowy	3 456,79																		
Procentowy	123,00%																		
Wykładniczy	3,46E+03																		
Maska wprowadzania	Ułatwia wprowadzanie danych przez określenie zestawu dostępnych znaków (litery lub cyfry) oraz elementów rozdzielających (np. nawiasy, myślniki). Program udostępnia kreatora masek wprowadzania dla zapisu typowych informacji: <table border="1" data-bbox="470 1597 933 1910"> <tbody> <tr> <td>Numer telefonu</td> <td>661-54-05</td> </tr> <tr> <td>Kod pocztowy</td> <td>99-878</td> </tr> <tr> <td>Numer NIP</td> <td>527-010-33-91</td> </tr> <tr> <td>Hasło</td> <td>*****</td> </tr> <tr> <td>Godzina długa</td> <td>00:00:00</td> </tr> <tr> <td>Data krótka</td> <td>1969-09-27</td> </tr> <tr> <td>Godzina krótka</td> <td>00:00</td> </tr> <tr> <td>Godzina średnia</td> <td>12:00</td> </tr> <tr> <td>Data średnia</td> <td>27-wrz-69</td> </tr> </tbody> </table> Można tworzyć własne maski wprowadzania z wykorzystaniem symboli:	Numer telefonu	661-54-05	Kod pocztowy	99-878	Numer NIP	527-010-33-91	Hasło	*****	Godzina długa	00:00:00	Data krótka	1969-09-27	Godzina krótka	00:00	Godzina średnia	12:00	Data średnia	27-wrz-69
Numer telefonu	661-54-05																		
Kod pocztowy	99-878																		
Numer NIP	527-010-33-91																		
Hasło	*****																		
Godzina długa	00:00:00																		
Data krótka	1969-09-27																		
Godzina krótka	00:00																		
Godzina średnia	12:00																		
Data średnia	27-wrz-69																		

Sym-bol	Opis
0	Cyfra (0 do 9, wpis wymagany, znaki plus [+] i minus [-] niedozwolone)
9	Cyfra lub odstęp (wpis niewymagany, dopuszczalne znaki + i -)
#	Cyfra lub odstęp (wpis niewymagany, odstępy w trybie edycji wyświetlane jako puste miejsca, które są usuwane przy zapisie danych, dopuszczalne znaki + i -)
L	Litera (A do Z, wpis wymagany)
?	Litera (A do Z, wpis opcjonalny)
A	Litera lub cyfra (wpis wymagany)
a	Litera lub cyfra (wpis opcjonalny)
&	Dowolny znak lub odstęp (wpis wymagany)
C	Dowolny znak lub odstęp (wpis opcjonalny)
. , ; - /	Znak oddzielenia dziesiętnego, separatory daty i czasu (zależne od ustawień regionalnych Windows)
<	Zamienia wprowadzone znaki na małe
>	Zamienia wprowadzone znaki na duże
!	Powoduje, że znaki w masce wprowadzania są wyświetlane od strony prawej do lewej zamiast od lewej do prawej
\	Kolejny znak jest wyświetlany jako literał (przykładowo, „\?” jest wyświetlane jako „?”)

Poniżej przedstawiono kilka przykładów masek wprowadzania i wartości danych

Maska wprowadzania	Przykładowe wartości
(000) 000-0000	(206) 555-0248
(999) 999-9999	(206) 555-0248
	() 555-0248
(000) AAA-AAAA	(206) 555-TELE
#999	-20
	2000
>L????L?000LO	GRENGR339M3
	MAY R 452B7
>LOL OLO	T2F 8M4
00000-9999	98115-
	98115-3007
>L<???????????????	Maria
	Bartłomiej
SSN 000-00-0000	SSN 555-55-5555
>LL? 00AAa	RZ 54909

RZE 66HY	
Wartość domyślna	Umożliwia podanie wartości automatycznie wprowadzanej do pola po dodaniu nowego rekordu. Przykład: w polu Data wystawienia faktury ustawiamy wartość domyślną Now() – funkcję podającą datę i czas wstawiania rekordu.
Reguła sprawdzania poprawności	Określenie wyrażenia, które musi być prawdziwe zawsze, gdy jest dodawana lub zmieniana wartość w tym polu. Do stosowania razem z właściwością <i>Tekst reguły sprawdzania poprawności</i> . Przykład: Data zatrudnienia pracownika musi być większa lub równa dacie utworzenia firmy. Jeśli firma została założona 2-sty-2011 to reguła w polu Data Zatrudnienia może wyglądać następująco: >2011-01-02
Tekst reguły spr. poprawności	Komunikat wyświetlany wtedy, gdy wprowadzona wartość nie spełnia <i>Reguły sprawdzania poprawności</i> . Kontynuując przykład tekst reguły sprawdzania poprawności może mieć brzmienie: <i>Wprowadź datę po 1 stycznia 2011</i>
Wymagane	Wymaganie wprowadzenia danych w polu. Dopuszczalne wartości tak lub nie.
Zerowa dł. dozwolona	Umożliwienie wprowadzenia ciągu o zerowej długości ("") w polu typu Hiperłącze, Tekst lub Nota przez ustawienie wartości <i>Tak</i> .
Indeksowane	Określenie, czy pole ma indeks (Funkcja, która przyspiesza wyszukiwanie i sortowanie danych w tabeli). Istnieją trzy dostępne wartości: <ul style="list-style-type: none"> • <i>Tak (Bez duplikatów)</i> — tworzy unikatowy indeks w polu. • <i>Tak (Duplikaty OK)</i> — tworzy nieunikatowy indeks w polu. • <i>Nie</i> — usuwa indeks z pola. <i>Uwaga</i> Tej właściwości nie należy zmieniać dla klucza podstawowy.
Kompresja Unicode	Kompresowanie tekstu przechowywanego w polu, gdy jest przechowywanych mniej niż 4 096 znaków.
Tryb IME	Sterowanie konwersją znaków we wschodnioazjatyckich wersjach systemu Windows.
Tryb zdania edytora IME	Sterowanie konwersją zdań we wschodnioazjatyckich wersjach systemu Windows.
Tagi inteligentne	Załączenie do pola tagu inteligentnego
Wyrównanie tekstu	Określenie domyślnego wyrównania tekstu w formancie.