



Odkryj świat baz danych  
z nową wersją Accessa

# Ćwiczenia praktyczne

# 2010 PL Access

Danuta Mendrala, Marcin Szeliga

- ▼ Poznaj **obiekty** baz danych
- ▼ Zaplanuj **tabele i relacje**
- ▼ Zaprojektuj **formularze**
- ▼ Zdefiniuj **kwerendy**
- ▼ Wygeneruj **raporty**



## » Idź do

- Spis treści
- Przykładowy rozdział

## » Katalog książek

- Katalog online
- Zamów drukowany katalog

## » Twój koszyk

- Dodaj do koszyka

## » Cennik i informacje

- Zamów informacje o nowościach
- Zamów cennik

## » Czytelnia

- Fragmenty książek online

## » Kontakt

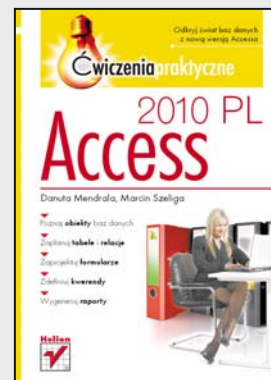
Helion SA  
ul. Kościuszki 1c  
44-100 Gliwice  
tel. 32 230 98 63  
e-mail: helion@helion.pl  
© Helion 1991–2010

## Access 2010 PL. Ćwiczenia praktyczne

Autorzy: [Danuta Mendrala](#), [Marcin Szeliga](#)

ISBN: 978-83-246-2617-5

Format: A5, stron: 192



### Odkryj świat baz danych z nową wersją Accessa

- Poznaj obiekty baz danych
- Zaplanuj tabele i relacje
- Zaprojektuj formularze
- Zdefiniuj kwerendy
- Wygeneruj raporty

Microsoft Access jest jednym z najpopularniejszych systemów zarządzania bazami danych na świecie, a jego najnowsza wersja z pewnością tylko ugruntuje pozycję lidera tej aplikacji na rynku. Access wykorzystywany jest zarówno przez użytkowników indywidualnych, jak i duże przedsiębiorstwa, których pracownicy doceniają nie tylko prostotę korzystania z interfejsu aplikacji, lecz także rozbudowane możliwości programowania i stosowania makr. Nie bez znaczenia jest tu też bogaty zestaw narzędzi i kreatorów, dzięki którym tworzenie baz, zarządzanie nimi i przetwarzanie staje się naprawdę łatwe, a przede wszystkim wydajne.

„Access 2010 PL. Ćwiczenia praktyczne” to wprowadzenie do świata baz danych i aplikacji bazodanowych tworzonych za pomocą najnowszej wersji Accessa. Treść książki i zawarte w niej praktyczne ćwiczenia pozwolą Ci sprawnie poznać podstawowe zasady korzystania z programu, projektowania baz danych i prawidłowego stosowania ich obiektów, a także właściwego tworzenia relacji. Nauczysz się też przygotowywać formularze i kwerendy ułatwiające zarządzanie danymi oraz generować raporty. Twoją wiedzę uzupełnią informacje na temat makr i modułów.

- Praca z plikami baz danych
- Tworzenie tabel i relacji
- Stosowanie kwerend
- Projektowanie formularzy
- Tworzenie raportów
- Makropolecenia
- Import i eksport danych do różnych formatów

**Skorzystaj z najnowszej wersji Accessa do tworzenia wydajnych aplikacji bazodanowych!**

# Spis treści

Wstęp	5
<b>Rozdział 1. Baza danych</b>	<b>9</b>
<b>Rozdział 2. Tabele</b>	<b>31</b>
<b>Rozdział 3. Relacje</b>	<b>55</b>
<b>Rozdział 4. Kwerendy</b>	<b>75</b>
Kwerendy wybierające dane	76
Kwerendy krzyżowe	94
Kwerendy tworzące tabele	96
Kwerendy aktualizujące	98
Kwerendy dołączające	99
Kwerendy usuwające	102
Ćwiczenia dodatkowe	104
<b>Rozdział 5. Formularze</b>	<b>107</b>
<b>Rozdział 6. Raporty</b>	<b>133</b>
<b>Rozdział 7. Makra</b>	<b>147</b>
<b>Rozdział 8. Moduły</b>	<b>157</b>



# Relacje



Charakterystyczną cechą relacyjnych baz danych jest przechowywanie informacji, opisujących obiekty różnego typu (np. towary i ich producentów), w odrębnych, ale powiązanych ze sobą tabelach. Na rysunkach 3.1 i 3.2 pokazujemy różnicę pomiędzy sposobem przechowywania tych samych danych w arkuszu programu Excel i w bazie danych programu Access.

	A	B	C	D	E	F	G	H
1	Imię	Nazwisko	Tytuł	Cena filmu	Zysk z wypożyczenia	Data Produkcji	Opinia	Gatunek
2	Jan	Machulski	51 stan	55,65	3,34	2001-03-23		Dramat
3	Jan	Machulski	5th Element	45,23	44,2	2002-02-16		Political Fictions
4	Jan	Machulski	O czym szumi wierzby	12	0	1999-12-23		S-F
5	Jan Jakub	Kolski	40 Days and 40 Nights	23,3	22	1999-12-23		Dramat
6	Jan Jakub	Kolski	The Meaning Of Life	145,8	456,33	1973-12-12	Klasyka, tr	Komedia
7	Jan Jakub	Kolski	The Life	123	23,9	1998-05-23		Komedia
8	Steven	Spielberg	6th Day	87,4	200,23	1998-12-30		Horror
9	Steven	Spielberg	War Games	32	0			Horror

**Rysunek 3.1.** Arkusz programu Excel jest przykładem jednorodnej bazy danych — takiej, w której wszystkie dane zapisane są w jednym obiekcie (arkuszu)

Nawet ten bardzo uproszczony przykład pozwala zauważyć główną zaletę relacyjnych baz danych — dzięki podzieleniu informacji między kilka tabel do minimum została zmniejszona liczba nadmiarowych, wielokrotnie przechowywanych informacji. O ile w arkuszu Excela dane reżysera pojawiły się tyle razy, ile nakręcił filmów, o tyle w bazie programu Access, niezależnie od liczby nakręconych przez reżysera filmów, jego dane zapisane są raz, a jedynie krótki identyfikator (klucz podstawowy) reżysera znajduje się przy każdym jego filmie.

The screenshot displays three tables in Microsoft Access:

- Reżyser** table:
 

IdReżysera	Imię	Nazwisko
1	Jan	Machulski
2	Jan Jakub	Koński
3	Steven	Spielberg
(Nowy)		
- Gatunek** table:
 

IdGatunku	Nazwa gatunku
1	S-F
2	Komedia
3	Horror
4	Dramat
- Film** table:
 

IdFilmu	Tytuł	Cena filmu	Zysk z wypożyczenia	Data Produkcji	Opinia	Gatunek	Reżyser
4	40 Days and 40 Nights	23,3	22	1999-12-23		4	2
3	51 stan	55,65	3,34	2001-03-23		4	1
4	5th Element	45,23	44,2	2002-02-16		7	1
7	6th Day	87,4	200,23	1998-12-30		3	3
9	The Meaning Of Life	145,8	456,33	1973-12-12	Klasyka, trzeci	2	2
10	The Life	123	23,9	1998-05-23		2	2
13	O czym szumi wierzby	12	0	1999-12-23		1	1
15	War Games	32	0			3	3

**Rysunek 3.2.** W relacyjnych bazach danych informacje o obiektach różnego typu (np. filmach, ich gatunkach i reżyserach) przechowywane są w odrębnych tabelach



Wielokrotne zapisywanie tych samych danych prędzej czy później prowadzi do problemów z utrzymaniem spójnego (adekwatnego do stanu rzeczywistego) obrazu danych — gdyby w naszym przykładzie reżyser zmienił nazwisko, zmiana ta musiałaby być przeprowadzona dla niewiadomej, potencjalnie dość dużej liczby filmów. W takiej sytuacji bardzo łatwo o błąd, wystarczy zmienić nazwisko w dziesięciu i zostawić niezmienione w pięciu komórkach.

Jednak użytkowników interesuje z reguły komplet informacji, np. chcieliby poznać tytuł filmu, dane jego reżysera i nazwę gatunku, do którego film należy. W relacyjnej bazie danych informacje te zapisane są w trzech odrębnych tabelach: w tabeli *Film* znajdują się dane o filmach, w tabeli *Reżyser* — dane reżyserów, a w tabeli *Gatunek* — nazwy gatunków filmowych. Aby informacja o tym, kto nakręcił dany film, nie została utracona, tabele te muszą być ze sobą powiązane.

Związek zachodzący pomiędzy powiązanimi ze sobą tabelami w SZBD Access nazywany jest relacją łączącą te tabele. Tak jak poszczególne kolumny tabeli przechowują wartości atrybutów (cech) obiektów danego typu (np. nazwę firmy, adres koleżanki czy tytuł książki), tak relacje przedstawiają zachodzący pomiędzy obiektami dwóch różnych typów związek (np. towar został sprzedany przez firmę, koleżanka pożyczyla płyty, autor napisał książkę).

Istnieją trzy typy relacji dwuargumentowych (czyli takich, które odpowiadają związkowi łączącemu dwie różne tabele). Oto one.

1. Relacje typu jeden do jednego, w których jednemu obiektowi A odpowiada dokładnie jeden obiekt B (przykładem takiej relacji jest związek pomiędzy uczniem a przypisaną mu szafką w szatni).
2. Relacje typu jeden do wielu, w których jednemu obiektowi A odpowiada dowolna liczba obiektów B (przykładem takiej relacji jest związek pomiędzy firmą a towarami — jedna firma może produkować dowolną liczbę towarów, ale ten sam towar może być wyprodukowany tylko w jednej firmie).
3. Relacje typu wiele do wielu, w których dowolnej liczbie obiektów A odpowiada dowolna liczba obiektów B (przykładem takiej relacji jest związek pomiędzy filmami a aktorami — w każdym filmie może zagrać dowolna liczba aktorów i każdy aktor może zagrać w dowolnej liczbie filmów).

Wykorzystując umiejętności zdobyte w poprzednich rozdziałach, utworzymy bazę danych działającą w środowisku SZBD Access, która będzie podstawą do ćwiczeń z następujących rozdziałów.

## Ć W I C Z E N I E

### 3.1 Projekt przykładowej bazy danych

Na potrzeby tej książki zaprojektujemy i utworzymy bazę danych *Filmoteka*, w której przechowywane będą podstawowe informacje o filmach i pożyczających je osobach. Zanim jednak zbudujemy nową bazę programu Access, należy zaprojektować układ jej tabel.

Aby zaprojektować przykładową bazę danych:

1. Wypisz, jedną pod drugą, nazwy typów obiektów, o których informacje powinny trafić do bazy:
  - Filmy
  - Osoby
2. Wypisz te atrybuty obiektów, o których informacje powinny być zapisane w bazie danych:
  - Film {Tytuł, Nazwisko reżysera, Imię reżysera, Nazwiska aktorów, Imiona aktorów, Gatunek, Cena, Data produkcji}
  - Osoba {Imię, Nazwisko, Adres, Telefon, Adres e-mail}
3. Na podstawie zdobytej wiedzy o zaletach dzielenia danych o obiektach różnego typu pomiędzy powiązane ze sobą tabele wyszukaj na liście wypisanych atrybutów te, które nie opisują bezpośrednio obiektu danego typu, i utwórz nową listę typów obiektów:

Film {Tytuł, Gatunek, Cena, Data produkcji}  
Osoba {Imię, Nazwisko, Adres, Telefon, Adres e-mail}  
Reżyser (Imię, Nazwisko) (trudno uznać imię czy nazwisko reżysera  
za bezpośrednie atrybuty filmu)  
Aktor (Imię, Nazwisko) (trudno uznać imię czy nazwisko aktora  
za bezpośrednie atrybuty filmu)

4. W poprzednim rozdziale omówione zostały zalety dzielenia informacji, opisujących obiekty poszczególnych typów, na atrybuty elementarne. Korzystając z tych informacji, zmodyfikuj listę atrybutów obiektów:

Film {Tytuł, Gatunek, Cena, Data produkcji}  
Osoba {Imię, Nazwisko, Miasto, Ulica Nr domu, Telefon, Adres e-mail}  
Reżyser (Imię, Nazwisko)  
Aktor (Imię, Nazwisko)

5. Jednym z etapów procesu normalizacji baz danych, który dodatkowo korzystnie wpłynie na wydajność bazy danych i komfort pracy użytkowników, jest wydzielenie tabel słownikowych. Tabele tego typu przechowują ustaloną liczbę danych wykorzystywanych do określania wartości pojedynczego atrybutu obiektu nadrzędnego takiej tabeli. W efekcie tabela nadrzędna jest mniejsza, a użytkownicy mogą wybrać daną wartość z listy, zamiast wielokrotnie wpisywać ten sam ciąg znaków. W przykładowej bazie danych idealnym kandydatem do roli tabeli słownikowej jest atrybut filmu — gatunek (lista gatunków filmowych jest skończona, a prawdopodobieństwo wielokrotnego wpisywania tych samych danych dość duże):

Film {Tytuł, Cena, Data produkcji}  
Gatunek {Nazwa}  
Osoba {Imię, Nazwisko, Miasto, Ulica Nr domu, Telefon, Adres e-mail}  
Reżyser (Imię, Nazwisko)  
Aktor (Imię, Nazwisko)

6. Ostatnim zadaniem jest określenie relacji łączącej poszczególne obiekty.
- Każdy film należy do jednego, określonego gatunku, ale do tego samego gatunku może należeć dowolna liczba filmów (związek typu wiele do jednego).
  - Każdy film został nakręcony przez jednego, określonego reżysera, ale ten sam reżyser może nakręcić dowolną liczbę filmów (związek typu wiele do jednego).

- W każdym filmie może zagrać dowolna liczba aktorów, a każdy aktor może zagrać w dowolnej liczbie filmów (związek typu wiele do wielu) — tę relację zaimplementujemy za pomocą charakterystycznych dla SZBD Access 2010 pól wielowartościowych.
- Każdy film (w tym samym czasie) może być pożyczony tylko jednej osobie, ale ta sama osoba może (jednocześnie) pożyczyć dowolną liczbę filmów (związek typu wiele do jednego) — ponieważ w bazie chcemy zapisać również datę pożyczania i oddania filmu, tę relację zaimplementujemy za pomocą tabeli łącznikowej.

## Ć W I C Z E N I E

**3.2 Tabela słownikowa**

Kolejność tworzenia tabel nie jest dowolna — jako pierwsze należy utworzyć tabele słownikowe, potem tabele podrzędne (tabele przechowujące dane, do których odwołują się inne tabele), a jako ostatnie — tabele nadrzędne (tabelą nadrzędną przykładowej bazy danych jest tabela *Film*, która odwołuje się do danych zapisanych we wszystkich pozostałych tabelach bazy danych).

Aby utworzyć tabelę *Gatunek*:

1. Utwórz nowy projekt pustej bazy danych programu Access i nazwij go *Filmoteka*.
2. Korzystając z umiejętności zdobytych w poprzednim rozdziale, utwórz tabelę *Gatunek*. Struktura tabeli została pokazana na rysunku 3.3.

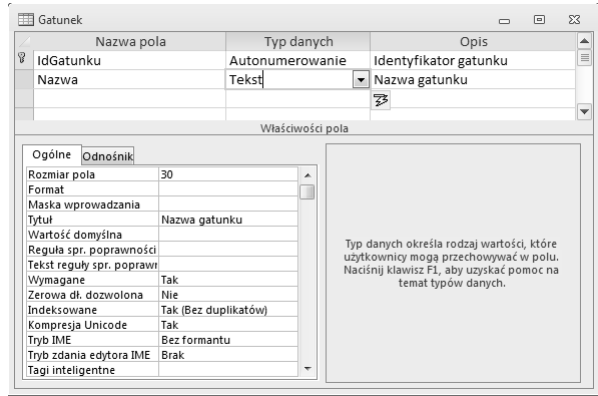
## Ć W I C Z E N I E

**3.3 Tabele podrzędne**

Po utworzeniu tabeli słownikowej przyszła pora na zbudowanie wszystkich tabel podrzędnych — ponieważ w przykładowej bazie danych nie zawierają one kolumn odwołujących się do innych tabel (nie ma ani jednej tabeli, która, będąc podrzędną w stosunku do jednej tabeli, jednocześnie byłaby nadrzędną w stosunku do innej tabeli), ich tworzenie nie różni się od opracowania tabel słownikowych.



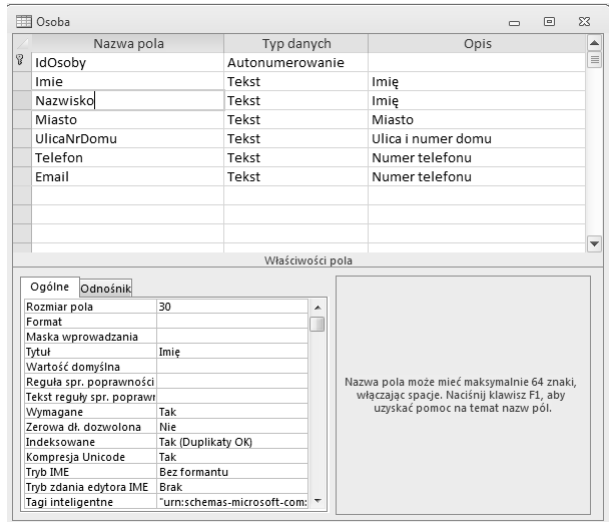
**Rysunek 3.3.**  
Definicja tabeli  
słownikowej



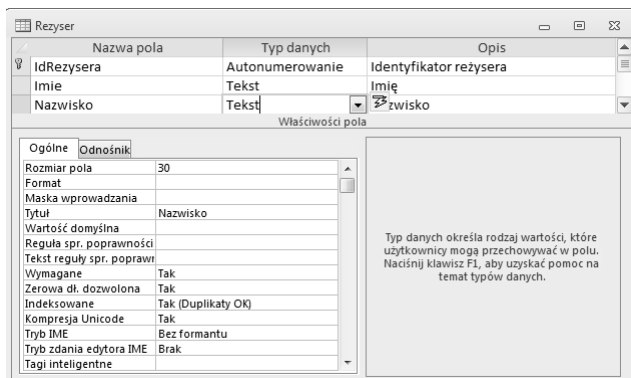
Aby utworzyć tabelę podrzędne (tabele niezawierające kolumn kluczy obcych):

1. Utwórz tabelę *Osoba* (rysunek 3.4).

**Rysunek 3.4.**  
Definicja tabeli  
podrzędnej  
*Osoba*



2. Utwórz tabelę *Rezyser*. Struktura tej tabeli została pokazana na rysunku 3.5.



**Rysunek 3.5.** Definicja tabeli *Rezyser* — tworząc tę (i pozostałe tabelę bazy *Filmoteka*) można wzorować się na tabelach dołączonej do książki bazy danych



W nazwach obiektów przykładowej bazy danych nie są używane polskie znaki diakrytyczne, natomiast w ich tytułach — tak. W ten sposób baza danych będzie prawidłowo działała w komputerach z zainstalowanym systemem Windows w innej niż polska wersji językowej.

3. Ponieważ struktura tabeli *Aktor* jest identyczna ze strukturą tabeli *Rezyser*, skopiuj tę tabelę pod nazwą *Aktor*, wyświetl nowo utworzoną tabelę w widoku projektu i zmień nazwę kolumny *IdRezysera* na *IdAktora*. Struktura pozostałych kolumn nie wymaga zmiany.

## Ć W I C Z E N I E

### 3.4 Tabela nadrzędna

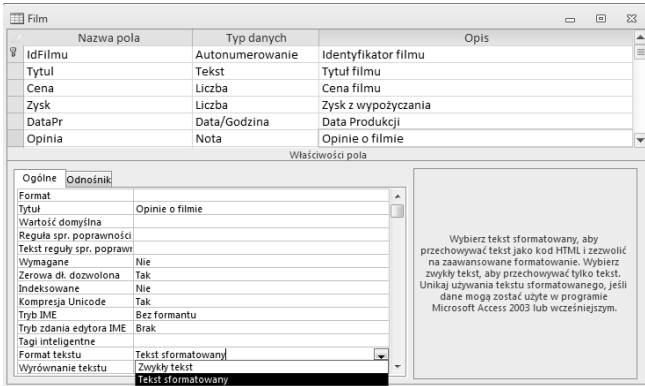
Na tym etapie baza danych *Filmoteka* zawiera cztery, niepowiązane ze sobą tabelę. Nadszedł czas, aby zamienić te niezwiązane ze sobą obiekty w relacyjną bazę danych, przechowującą informacje o poszczególnych filmach. Ponieważ tabela *Film* zawiera kolumny, wskazujące na dane zapisane w innych tabelach, jej utworzenie wymagać będzie zbudowania relacji łączących odpowiednie tabelę.

Aby utworzyć tabelę *Film*:

1. Rozpocznij tworzenie nowej tabeli w widoku projektu.
2. Utwórz kolumnę klucza podstawowego (*IdFilmu*) oraz kolumny przechowujące informacje o tytule danego filmu (*Tytuł*), cenie

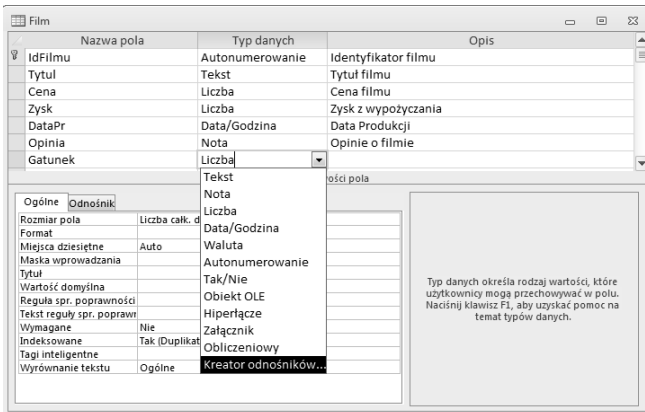
zakup (Cena), zysku, jaki uzyskaliśmy z wypożyczenia danego filmu (Zysk), i dacie produkcji (DataPr).

3. Jedną z cech charakterystycznych dla SZBD Access 2010 jest możliwość formatowania danych typu *Nota*. Dodaj kolejną kolumnę *Opinia* typu *Nota* i zmień wartość atrybutu *Format tekstu* na *Tekst sformatowany* (rysunek 3.6).



**Rysunek 3.6.** Wpisując opinie o filmach, będziemy mogli dowolnie zmieniać krój, wielkość i kolor czcionki

4. Ustaw kursor w polu *Typ danych* nowej kolumny i z listy dostępnych typów wybierz *Kreator odnośników...* (rysunek 3.7).

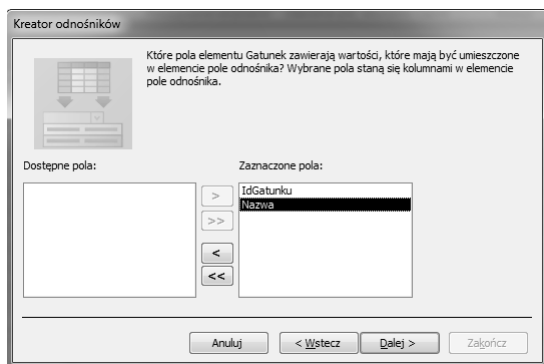


**Rysunek 3.7.** Aby uniknąć kłopotów z niezgodnościami typów łączonych kolumn różnych tabel, należy najpierw uruchomić kreator odnośników, a później (ewentualnie) zmienić nazwę kolumny

5. Ponieważ chcemy połączyć tabelę *Film* z tabelą *Gatunek*, w odpowiedzi na pierwsze pytanie kreatora wybierz opcję *Obiekt kolumna odnośnika ma pobierać wartości z tabeli lub kwerendy*.
6. Na liście dostępnych tabel zaznacz tabelę *Gatunek* i kliknij przycisk *Dalej*.
7. Zostanie wyświetlona lista kolumn tabeli *Gatunek*. Chociaż table zawsze powinny być łączone za pomocą ich kluczy podstawowych (w tym przypadku będą to wartości pól kolumny *IdGatunku*), identyfikatory poszczególnych gatunków są mało opisowe (administrator musiałby zapamiętać, że np. identyfikatorem gatunku horror jest liczba 12, a komedii — 3). SZBD Access ułatwia administratorowi przeglądanie i modyfikowanie danych w połączonych tabelach, bo automatycznie zastępuje wyświetlany identyfikator związanymi z nim wartościami innych pól danej tabeli. I tak zamiast wpisywać liczbę 3 w oknie widoku arkusza danych tabeli, będziemy mogli wybrać nazwę gatunku danego filmu. Aby osiągnąć ten cel, zaznacz oba pola tabeli *Gatunek* i kliknij przycisk *Dalej* (rysunek 3.8).

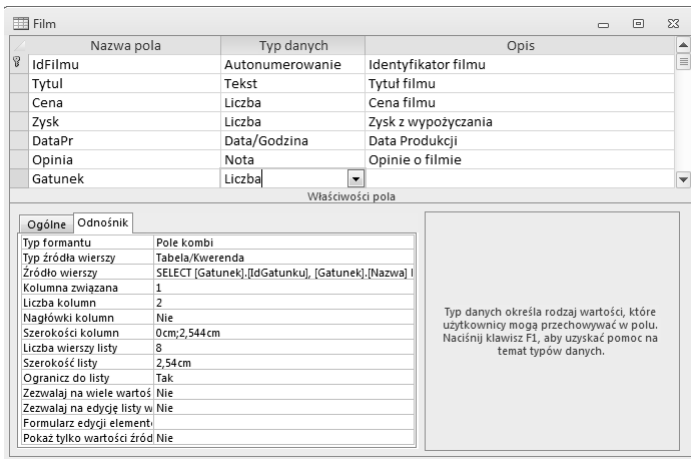
**Rysunek 3.8.**

*Kreator odnośników umożliwia „ukrycie” przed administratorem wykorzystanego do połączenia tabel klucza głównego i zastąpienie go dowolną, opisową wartością, np. nazwą danego gatunku filmowego*



8. SZBD Access 2010 umożliwia też posortowanie danych wykorzystywanych do łączenia tabel. Aby ułatwić administratorowi przypisywanie poszczególnych filmów do gatunków, posortuj je alfabetycznie według nazw.

9. Odpowiadając na kolejne pytanie kreatora, upewnij się, czy pole wyboru *Ukryj kolumnę klucza (zalecane)* jest zaznaczone, i kliknij przycisk *Dalej* — w ten sposób gatunki filmowe będą wybierane wyłącznie na podstawie nazw.
10. Ostatnie pytanie kreatora będzie dotyczyło nazwy tworzonej kolumny:
  - Wpisz taką samą nazwę, jaką ma łączona tabela (Gatunek),
  - Nie zaznaczaj pola *Włącz integralność danych* (jego znaczenie wyjaśniliśmy w ćwiczeniu poświęconym edytowaniu relacji),
  - Nie zaznaczaj pola *Zezwalaj na wiele wartości* (jego znaczenie wyjaśniliśmy w dalszej części rozdziału na przykładzie relacji łączącej tabele *Film* i *Aktor*),
  - I kliknij przycisk *Zakończ*.
11. Po zakończeniu pracy kreatora odnośników konieczne będzie zapisanie tabeli nadrzędnej — zapisz ją pod nazwą *Film*. Wynik pracy kreatora został pokazany na rysunku 3.9.



**Rysunek 3.9.** Zakładka *Odnosińnik* pozwala poznać efekt pracy kreatora odnośników: w tym przypadku dane pobierane z tabeli *Gatunek* będą wyświetlane w polu listy do połączenia tabel wykorzystana została pierwsza kolumna (kolumna klucza podstawowego), ale dane będą pobierane z dwóch kolumn. Ponieważ szerokość pierwszej kolumny została ustawiona na 0, użytkownik zobaczy w polu listy wyłącznie dane przechowywane w drugiej kolumnie tabeli *Gatunek*

12. W ten sam sposób, w jaki połączyliśmy tabele *Film* i *Gatunek*, należy połączyć tabele *Film* i *Reżyser*:
- ❑ Po ustawieniu kursora w pustym polu kolumny *Typ danych* z listy dostępnych typów wybierz *Kreator odnośników...*
  - ❑ Ponownie dane będą pobierane z innej tabeli, ale tym razem z tabeli *Reżyser*.
  - ❑ Aby ułatwić przeglądanie i wprowadzanie danych bezpośrednio do tabeli, z listy kolumn tabeli *Aktor* należy wybrać *IdReżysera* (klucz podstawowy powinien być wykorzystany do łączenia wszystkich tabel) oraz *Nazwisko* (istnieje mniejsze prawdopodobieństwo, że kilku reżyserów będzie nosić to samo nazwisko niż to samo imię).



Możliwe jest również wskazanie większej liczby kolumn z tabeli połączonej, ale wtedy dane wyświetlane w polu listy będą zajmować dużo miejsca na ekranie i w efekcie, zamiast ułatwiać, będą utrudniać korzystanie z bazy danych.

- ❑ Po posortowaniu danych według nazwisk aktorów i ukryciu pola klucza podstawowego pozostaje jeszcze podanie nazwy tworzonej kolumny odnośnika (*Reżyser*) i zapisanie zmian w tabeli *Film*.



Odpowiednikiem utworzenia odnośnika do innej tabeli jest nałożenie na kolumnę ograniczenia REFERENCES (klucz obcy). W efekcie ograniczymy listę dopuszczalnych dla danej kolumny wartości do wartości przechowywanych w odpowiadającej jej kolumnie w powiązanej tabeli. Kolumny z nałożonym ograniczeniem REFERENCES mogą przechowywać wartości NULL (niewypełnione pola), o ile nie zabraniają tego inne ograniczenia. Aby SZBD Access mógł utworzyć klucz obcy, musi wcześniej być utworzona tabela, do której klucz będzie się odwoływał. Dodatkowo w tabeli tej musi znajdować się odpowiednia kolumna (lub kolumny), dla której zdefiniowano taki sam typ danych jak dla kolumny z nałożonym ograniczeniem REFERENCES. Na kolumnę taką musi być także nałożone jedno z dwóch ograniczeń: PRIMARY KEY lub UNIQUE.

## Ć W I C Z E N I E

**3.5 Pola wielowartościowe**

Skoro w jednym filmie gra wielu aktorów, a ten sam aktor występuje w wielu filmach, wydaje się, że niemożliwe jest utworzenie kolumny odnośnika w tabeli *Film* (wtedy w filmie mógłby zagrać tylko jeden aktor) ani w tabeli *Aktor* (wtedy z kolei aktor mógłby zagrać tylko w jednym filmie). Ponieważ wiele osób próbowało — błędnie — rozwiązać ten problem, dodając do jednej z tabel kilka kolumn (w naszym przykładzie do tabeli *Film* mogliśmy dodać kolumny *Aktor1*, *Aktor2* itd.), SZBD Access uproszczył tworzenie relacji typu wiele do wielu.



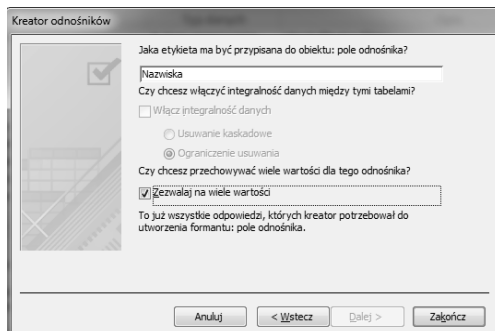
Umieszczenie w tabeli wielu kolumn, które odwołują się do tej samej kolumny innej tabeli, jest bardzo złym pomysłem. Nie tylko znacznie ogranicza funkcjonalność bazy (musimy z góry założyć maksymalną liczbę aktorów, którzy mogą zagrać w jednym filmie) i powoduje, że w naszej tabeli przechowywanych jest mnóstwo wartości Null (bo w większości przypadków interesuje nas tylko kilku aktorów grających w danym filmie), ale również ogromnie utrudnia tworzenie powiązanych z taką tabelą formularzy i raportów.

Aby połączyć tabele *Film* i *Aktor*:

1. Wyświetl projekt tabeli *Film*.
2. Dodaj kolejną kolumnę odnośnika, wiążąc ją z kolumnami *IdAktora* i *Nazwisko* tabeli *Aktor*.
3. Posortuj dane według nazwisk i ukryj kolumnę klucza.
4. Kończąc pracę kreatora:
  - zmień nazwę kolumny odnośnika na *Nazwiska*,
  - zaznacz pole *Zezwalaj na wiele wartości* (rysunek 3.10).

**Rysunek 3.10.**

*W ten sposób w kolumnie odnośnika Nazwiska będzie można zapisywać dowolną liczbę nazwisk grających w danym filmie aktorów*



5. Zamknij okno projektu tabeli *Film* i zapisz wprowadzone w niej zmiany.

## Ć W I C Z E N I E

### 3.6 Edytowanie relacji

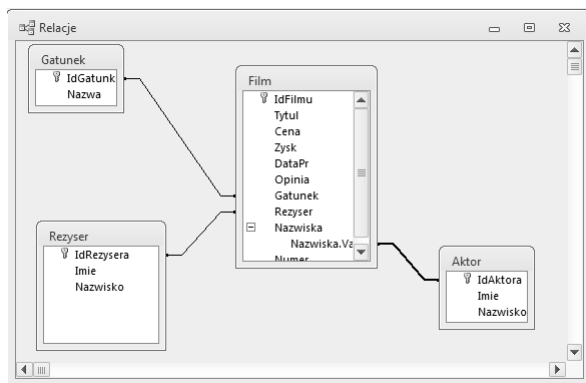
Zanim połączymy table *Film* i *Osoba*, zmodyfikujemy relacje utworzone między pozostałymi tabelami.

W tym celu:

1. Przejdź na zakładkę *Narzędzia bazy danych* i kliknij przycisk *Relacje*.
2. Aby upewnić się, że wszystkie połączone table zostały wyświetlone, kliknij prawym przyciskiem myszy pusty obszar okna relacji i z menu kontekstowego wybierz opcję *Pokaż wszystko*.
3. Za pomocą techniki „przeciągnij i upuść” uporządkuj ikony tabel, tak aby linie obrazujące relacje nie krzyżowały się, i dopasuj rozmiar ikon poszczególnych tabel, tak by widoczne były wszystkie ich kolumny (rysunek 3.11).

**Rysunek 3.11.**

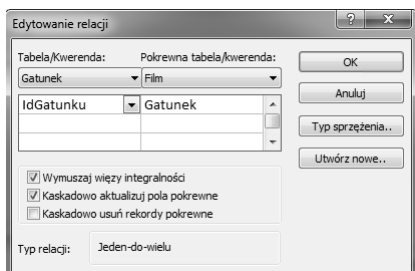
*Linie łączące poszczególne table przedstawiają utworzone pomiędzy nimi powiązania*



4. Na tym etapie relacje łączące poszczególne table nie odzwierciedlają jeszcze typów powiązań (wszystkie te powiązania powinny być typu jeden do wielu). Aby określić typ relacji:



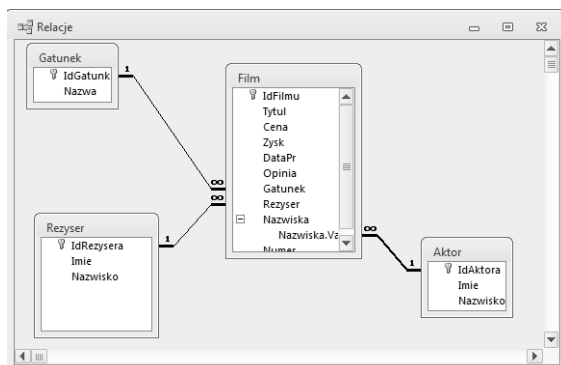
- ❑ Kliknij prawym przyciskiem myszy linię łączącą tabele *Film* i *Gatunek*, a następnie z menu kontekstowego wybierz opcję *Edytuj relację...*
- ❑ Zaznacz pole wyboru *Wymuszaj więzy integralności* — w ten sposób określony zostanie typ powiązania i SZBD Access będzie automatycznie sprawdzał, czy dane w połączonych tabelach są prawidłowe (np. czy użytkownik nie próbował przypisać filmu do nieistniejącego w bazie gatunku).
- ❑ Zaznacz pole wyboru *Kaskadowo aktualizuj pola pokrewne* — w ten sposób zmiana wartości klucza podstawowego tabeli podrzędnej zostanie automatycznie uwzględniona w tabeli nadrzędnej (jeżeli ktoś zmieni identyfikator gatunku filmowego, np. komedii z 3 na 33, zmiana ta zostanie automatycznie przeprowadzona w tabeli *Film*, dzięki czemu filmy, które wcześniej zostały zaklasyfikowane jako komedie, nadal będą komediami) (rysunek 3.12).



**Rysunek 3.12.** O ile kaskadowa aktualizacja powiązanych pól zabezpiecza przed przypadkową modyfikacją wartości kluczy podstawowych, o tyle kaskadowe usuwanie może spowodować przypadkowe usunięcie wielu danych

- ❑ Nie zaznaczaj pola wyboru *Kaskadowo usuń rekordy pokrewne* — zaznaczenie tego pola umożliwi nie tylko usunięcie gatunku, do którego zostały zaklasyfikowane pewne filmy, ale SZBD Access automatycznie wyrzuci z bazy informacje o wszystkich filmach usuwanego gatunku (jeżeli ktoś usunąłby wiersz z gatunkiem komedia, z tabeli *Film* zostałyby usunięte wszystkie filmy zaklasyfikowane jako komedie).
- ❑ Kliknij przycisk *OK*.

- Linia łącząca tabele *Film* i *Gatunek* zostanie pogrubiona i, co najważniejsze, na jej końcach wyświetlone zostaną symbole  $1$  i  $\infty$ , obrazujące typ powiązania.
- Powtórz czynności opisane w punkcie 4. dla pozostałych relacji. Efekt pomyślnego wykonania ćwiczenia pokazany został na rysunku 3.13.



**Rysunek 3.13.** Zgodnie z założeniem projektu, jeden aktor (symbol 1) może zagrać w wielu (symbol  $\infty$ ) filmach, do jednego gatunku może należeć wiele filmów, jeden reżyser może nakręcić wiele filmów, a pole wielowartościowe pozwala wielu aktorom zagrać w tym samym filmie

- Zamknij okno relacji i zapisz wprowadzone zmiany.

## Ć W I C Z E N I E

### 3.7 Wprowadzenie przykładowych danych

Po zdefiniowaniu łączących tabele relacji należy wprowadzić przykładowe dane do tabel słownikowych i podrzędnych.

Aby wprowadzić przykładowe dane do bazy:

- Wyświetl tabelę *Gatunek* w widoku arkusza danych.
- Wpisz nazwy kilku gatunków filmów.
- Ponieważ nie planujemy podczas wpisywania danych dodawania do tabel bazy *Filmoteka* dodatkowych kolumn, należy ukryć kolumnę *Dodaj nowe pole*:
  - rozwij menu *Plik* i kliknij przycisk *Opcje*,
  - przejdź na zakładkę *Bieżąca baza danych*,

- ❑ usuń zaznaczenie pola wyboru *Włącz zmiany projektu w tabelach w widoku arkusza danych*,
  - ❑ kliknij *OK* (żeby zmiany zaczęły obowiązywać, konieczne może być zamknięcie i ponowne otwarcie bazy danych).
4. Aby przekonać się, czy możliwe jest wpisanie filmów bez jednoczesnego określenia gatunku, reżysera czy aktora, wprowadź przykładowe dane o kilku filmach. Zwróć uwagę na możliwość formatowania wpisywanych opinii (rysunek 3.14).

Identyfikator	Tytuł	Cena filmu	Zysk z wypożyczania	Data Produkcji	Opinie o filmie
2	40 Days and 40 Nights	23,3	22	1999-12-23	
3	51 stan	55,65	3,34	2001-03-23	
4	5th Element	45,23	44,2	2002-02-16	
7	6th Day	87,4	200,23	1998-12-30	
9	The Meaning Of Life	145,8	456,33	1973-12-12	Klasyka, <i>trzeba zobaczyć</i>
10	The Life	123	23,9	1998-05-23	
13	O czym szumi wierzby	12	0	1999-12-23	

**Rysunek 3.14.** Przykładowe dane wprowadzone do tabeli *Film*

5. Przekonaj się, że chociaż pole odnośnika może zawierać wartość nieokreśloną (wartość Null), nie może zawierać wartości nieodpowiadającej jednemu z kluczy podstawowych powiązanej tabeli. Innymi słowy, niemożliwe jest zaklasyfikowanie filmu do nieistniejącego gatunku.
6. Zamknij okno tabeli *Film*.
7. Otwórz w widoku arkusza danych tabelę *Reżyser*. Wpisz nazwiska i imiona kilku reżyserów.
8. Zamknij okno tabeli *Reżyser* i otwórz w widoku arkusza danych tabelę *Aktor*. Wpisz dane kilku aktorów.
9. Zamknij okno tabeli *Aktor* i raz jeszcze otwórz w widoku arkusza danych tabelę *Film*. Ustaw kursor w polu *Gatunek* pierwszego z wpisanych filmów i kliknij wyświetloną z prawej strony pola strzałkę — zobaczysz listę nazw wprowadzonych gatunków filmów (rysunek 3.15).
10. Zwróć uwagę, że niemożliwe jest dodawanie nowych rekordów do tabel podrzędnych w widoku arkusza danych tabeli nadrzędnej. Aby np. dodać nowy gatunek filmów, należy otworzyć tabelę *Gatunek*, nie *Film*.

Tytuł	Cena filmu	Opinie o filmie	Gatunek filmu	Reżyser fi
40 Days and 40 Nights	23,3		Dramat	Kolski
51 stan	55,65		Dramat	Machulski
5th Element	45,23		12	Machulski
6th Day	87,4		Dokument	Spielberg
The Meaning Of Life	145,8	Klasyka, <i>trzeba zobaczyć</i>	Dramat	Kolski
The Life	123		Horror	Kolski
O czym szumi wierzby	12		Komedia	Machulski
War Games	32		Political Fictions	Spielberg

**Rysunek 3.15.** Gdy mamy już dane w tabelach podrzędnych, wpisując dane do tabeli nadrzędnej, możemy się do nich odwoływać. W tym przykładzie niektóre kolumny zostały ukryte — aby ukryć wybrane kolumny, należy kliknąć ich nagłówek prawym przyciskiem myszy i z menu kontekstowego wybrać opcję *Ukryj pola*

## Ć W I C Z E N I E

### 3.8 Tabela łącznikowa

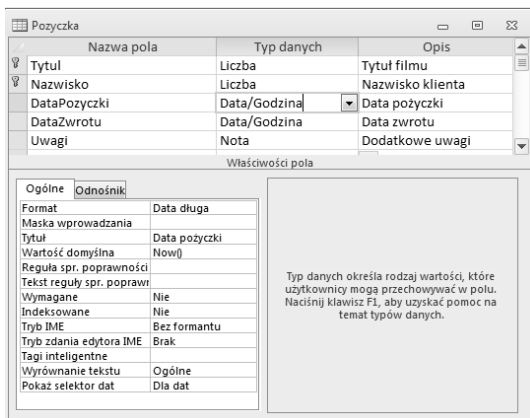
Tabele *Film* i *Osoba* połączone są związkiem typu wiele do wielu. Ponieważ chcemy zapisać pewne informacje (takie jak daty pożyczenia i zwrotu) o samym fakcie wypożyczenia filmu, implementacja tego związku wymaga utworzenia dodatkowej tabeli łącznikowej, w której przechowywane będą wartości kluczy podstawowych obu tabel. W ten sposób relacja typu wiele do wielu zostanie zastąpiona dwoma równorzędnymi relacjami typu wiele do jednego.

Aby zbudować tabelę łącznikową:

1. Utwórz nową tabelę w widoku projektu.
2. Za pomocą kreatora odnośników utwórz relację łączącą tę tabelę z tabelą *Film* (tak jak poprzednio, tabele muszą być powiązane na podstawie wartości klucza podstawowego tabeli podrzędnej, ale wyświetlane powinny być alfabetycznie ułożone tytuły filmów). Ta kolumna nie powinna przechowywać wielu wartości.
3. Zapisz nowo tworzoną tabelę pod nazwą *Pozyczka*.  
W odpowiedzi na ostrzeżenie o braku zdefiniowanego klucza podstawowego kliknij przycisk *Nie* — tabela łącznikowa, przechowująca wyłącznie wartości kluczy podstawowych powiązanych tabel, nie wymaga dodatkowej kolumny klucza podstawowego.

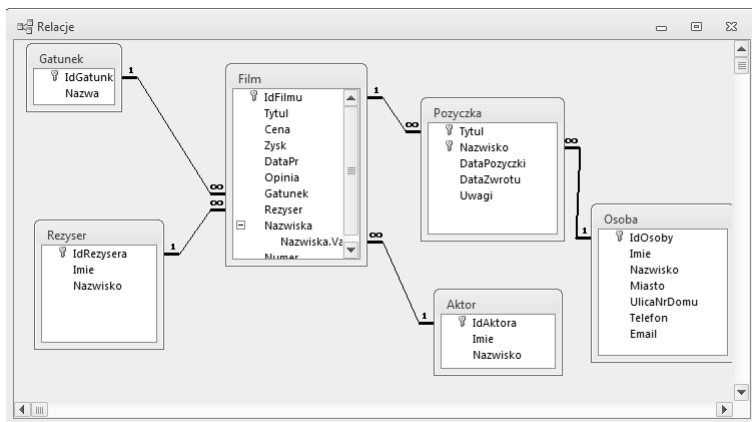
4. Za pomocą kreatora odnośników utwórz relację łączącą tę tabelę z tabelą *Osoba* (tak jak poprzednio, tabele muszą być powiązane na podstawie wartości klucza podstawowego tabeli podrzędnej, ale wyświetlane powinny być alfabetycznie ułożone nazwiska osób). Również pola tej kolumny nie będą przechowywały wielu wartości.
5. Dodaj kolumny *DataPożyczki* i *DataZwrotu* typu *Data/godzina* i o formacie *Data długa*. Nadaj obu kolumnom odpowiednie tytuły, a dla kolumny *DataPożyczki* dodatkowo określ wartość domyślną — wynik funkcji *Now* (w ten sposób datą pożyczania automatycznie będzie bieżąca data).
6. Dodaj jeszcze kolumnę *Uwagi* typu *Nota* i zezwól na formatowanie uwag dotyczących pożyczania przez daną osobę wybranego filmu.
7. Na końcu zaznacz dwie pierwsze kolumny tabeli (obie kolumny odnośników) i kliknij widoczny na wstążce przycisk *Klucz podstawowy* — w ten sposób kluczem tabeli *Pozyczka* będzie kombinacja identyfikatorów filmu i osoby (rysunek 3.16).

**Rysunek 3.16.**  
 Gotowy projekt tabeli łącznikowej (tabeli przechowującej informacje o pożyczonych różnym osobom filmach)



8. Zamknij okno projektu tabeli *Pozyczka* i wyświetl okno *Relacje*.
9. Kliknij prawym przyciskiem myszy wolny obszar okna relacji i z menu kontekstowego wybierz opcję *Pokaż wszystko* — wyświetlone zostaną tabele *Pozyczka* i *Osoba*.

10. Wykorzystaj umiejętności zdobyte w ćwiczeniu 3.5 i zmień relacje łączące tabele *Film*, *Pozyczka* i *Osoba* tak, aby SZBD wymuszał przestrzeganie więzów integralności, kaskadowo aktualizował i kaskadowo usuwał powiązane pola (rysunek 3.17).



**Rysunek 3.17.** Widok struktury bazy danych FilMOTEKA — taki układ tabel nazywany jest diagramem E/R (Encja/Relacja)

## Ć W I C Z E N I E

### 3.9 Korzystanie z tabeli łącznikowej

Dzięki tabeli łącznikowej *Pozyczka* w bazie danych możliwe będzie przechowywanie informacji o tym, które osoby wypożyczyły dane filmy, przy czym nie ma żadnych ograniczeń związanych z liczbą filmów, jakie może pożyczyć ta sama osoba, i liczbą osób, mogących wypożyczyć każdy film — tabele łącznikowe są powszechnie stosowane do tworzenia relacji typu wiele do wielu.

Aby wprowadzić do bazy informacje o wypożyczeniu i zwrocie filmów:

1. Otwórz w widoku arkusza danych tabelę *Osoba* i wpisz dane znajomych.
2. Zamknij okno tabeli *Osoba* i otwórz w widoku arkusza danych tabelę *Pozyczka*.

3. Kliknij pole *Tytuł* i z listy tytułów wybierz przykładowy tytuł filmu.
4. Przekonaj się, że niemożliwe jest zapisanie danych o wypożyczeniu filmu nieokreślonej osobie, tak samo jak niemożliwe jest wypożyczenie wskazanej osobie nieokreślonego filmu — Access nie pozwala na wpisywanie wartości Null do kolumn klucza podstawowego tabeli.
5. Ustaw kursor w polu *Nazwisko* i z listy nazwisk wybierz nazwisko znajomego.
6. Zwróć uwagę, że data wypożyczenia została automatycznie ustalona na dzisiejszą. Wpisanie daty zwrotu będzie oznaczało oddanie filmu.
7. Wpisz przykładowe dane do tabeli. Zwróć uwagę, że:
  - ten sam film może być wielokrotnie wypożyczony,
  - ta sama osoba może pożyczyć dowolną liczbę filmów,
  - daty, zamiast wpisywać, można wybierać z kalendarza,
  - uwagi mogą być dowolnie formatowane (rysunek 3.18).

The screenshot shows the 'Pozyczka' table in Access 2010. The table has five columns: 'Tytuł filmu', 'Nazwisko', 'Data pożyczki', 'Data zwrotu', and 'Dodatkowe uwagi'. The data rows are as follows:

Tytuł filmu	Nazwisko	Data pożyczki	Data zwrotu	Dodatkowe uwagi
40 Days and 40 Nights	Kowal	7 stycznia 2007	28 stycznia 2007	
40 Days and 40 Nights	Malicki	29 stycznia 2007		
51 stan	P.	10 stycznia 2007		
51 stan	Kowal	7 stycznia 2007	8 stycznia 2007	!!#@#!%
5th Element	Kowal	9 stycznia 2007		
6th Day	Malicki	9 styczn		
The Meaning Of Life	P.	7 styczn		
The Life	P.	12 styczn		
The Life	Kowal	9 styczn		
The Life	Malicki	10 styczn		
O czym szumi wierzby	P.	7 styczn		
War Games	Kowal	7 styczn		
War Games	Malicki	20 styczn		

A calendar pop-up is visible over the 'Data zwrotu' column, showing the month of May 2010. The date '14' is selected. The status bar at the bottom shows 'Rekord: 5 z 13' and 'Wyszukaj'.

**Rysunek 3.18.** Tabela łącznikowa umożliwiająca odzwierciedlanie związków typu wiele do wielu