

1. Podstawowe pojęcia związane z bazami danych
2. Etapy tworzenia bazy danych
3. Na czym polega relacja w bazie danych?
4. Określenie wymagań i ustalenie zbiorów informacji
5. Tabele
 - 5.1. Zasady tworzenia tabel
 - 5.2. Dlaczego bazy danych tworzy się w kilku tabelach?
 - 5.3. Tworzenie tabel
 - 5.4. Definiowanie relacji



Warto powtórzyć

1. Jakiego znasz zastosowania baz danych?
2. Podaj przykładowe operacje, które można wykonywać na danych w bazie danych.

1. Podstawowe pojęcia związane z bazami danych

Z **bazami danych** spotykamy się w wielu miejscach, np. takich, jak: sekretariat, sklep, apteka, magazyn, hurtownia, kino, teatr, bank, poczta, wypożyczalnia filmów, gabinet lekarski. Bazy danych mogą być wykorzystywane np. do obsługi zawodów sportowych czy obsługi wyborów – znalazły zastosowanie w prawie każdej dziedzinie życia.



Baza danych to zbiór danych, czyli odpowiednio **zorganizowanych informacji**, które można **przetwarzać** za pomocą programów komputerowych.

Korzystając z wybranej bazy danych, użytkownik komputera może wprowadzać do systemu komputerowego różne informacje (np. o uczniach, zawodnikach, lekach, książkach, produktach), które po wprowadzeniu do komputera (czyli zapisaniu w postaci cyfrowej) stają się **danymi**, a następnie mogą być **przetwarzane**.

Pojęcia **baza danych** używamy często także w odniesieniu do oprogramowania umożliwiającego tworzenie baz danych i przetwarzanie danych.



Przetwarzanie danych to uporządkowane wykonywanie operacji na **danych**, m.in.:

- wprowadzenie,
- modyfikowanie (poprawianie, uzupełnianie, aktualizowanie),
- wyszukiwanie,
- sortowanie,
- prezentowanie (wyświetlanie, drukowanie).

Przykładową relacyjną bazę danych i sposób tworzenia nowej bazy pokażemy na przykładzie programu Microsoft Access. Profesjonalne bazy danych tworzone są najczęściej z wykorzystaniem innego oprogramowania.



Dane w relacyjnej bazie danych są zorganizowane w **tabelach** reprezentujących logiczne powiązanie danych (**relacje**).

Tabela składa się z **wierszy** (zwanymi **rekordami**) i **kolumn** (zwanymi **polami**).

W jednym wierszu (rekordzie) znajdują się informacje o jednym obiekcie bazy danych, a kolumny (pola) zawierają wartości cech **atrybutów** tego obiektu.

Każda kolumna (każde pole) posiada unikalną w ramach tabeli nazwę.

Dla każdego pola można określić typ danych, które będą w nim przechowywane, np. **Tekst**, **Liczba**, **Data/Godzina**. Program Microsoft Access domyślnie przypisuje polom tekstowy typ danych (w praktyce większość gromadzonych danych jest typu tekstowego).



Przykład 1. Przeglądanie rekordów tabeli bazy danych

Utworzono bazę danych *Stadnina*, która zawiera dwie tabele: *Konie* (zob. rys. 1a) i *Wyścigi* (zob. rys. 1b). Tabela *Konie* zawiera informacje o koniach z pewnej stadniny: numer konia, jego imię, rasę, kraj pochodzenia, wiek, płeć, maść.

Polom nadano odpowiednio nazwy: *NrKonia*, *Imię*, *Rasa*, *KrajPochodzenia*, *Wiek*, *Płeć*, *Maść*.

Przykładowy rekord widoczny na rys. 1.: (2, *Demeter*, *arabska*, *Polska*, 4, *klacz*, *gniada*).

NrKonia	Imię	Rasa	KrajPochodzenia	Wiek	Płeć	Maść
1	Arbat	branderburska	Polska	3	ogier	kasztanowata
2	Demeter	arabska	Polska	4	klacz	gniada
3	Decybel II	hanowerska	Polska	4	klacz	kasztanowata
4	Vancouver	branderburska	Wielka Brytania	6	klacz	gniada
5	Vika	małopolska	Polska	4	ogier	tarantowata
6	Falaris	arabska	Polska	6	ogier	siwa
7	Galax	oldenburska	Niemcy	5	ogier	kara
8	Pianissima	arabska	Polska	4	klacz	gniada
9	Ekstern	arabska	Zjednoczone Emiraty Arabskie	6	klacz	siwa
10	Akcent	hanowerska	USA	3	ogier	gniada
11	Wiaźma	arabska	Wielka Brytania	4	ogier	gniada
12	First des Termes	hanowerska	Polska	3	klacz	siwa
13	Jowisz	huculska	Francja	6	ogier	siwa
14	Mila	huculska	Niemcy	3	klacz	gniada
15	Galerius	trakeńska	USA	9	klacz	kara
16	Pianosa	arabska	Zjednoczone Emiraty Arabskie	5	ogier	gniada

Rys. 1a. Tabela Konie (Microsoft Access)



Ćwiczenie 1.

Korzystając z rys. 1b (str. 106), opowiedz, jakie informacje są zawarte w tabeli *Wyścigi*? Podaj nazwy pól i przykładowe wartości zapisane w wyróżnionym rekordzie.

NrKonia	DataWyścigów	Miejscowość	NrGonitwy	Miejsce	Jeździec
1	2012-06-03	Pardubice	5	0	Forneka
1	2012-06-17	Wrocław	1	4	Faryska
1	2012-06-17	Wrocław	2	4	Popotko
1	2012-06-17	Wrocław	3	2	Wilich
1	2012-06-17	Wrocław	4	3	Kowalczyk
1	2012-06-17	Wrocław	5	1	Forneka
1	2012-06-24	Wrocław	1	5	Faryska
1	2012-06-24	Wrocław	2	1	Hemzel
1	2012-06-24	Wrocław	3	2	Popotko
2	2012-06-24	Warszawa	3	4	Hemzel
2	2012-06-24	Warszawa	4	4	Forneka
2	2012-06-24	Warszawa	5	2	Kowalczyk
2	2012-06-03	Pardubice	1	1	Wilich
2	2012-06-03	Pardubice	2	3	Tlarska
2	2012-06-03	Pardubice	3	4	Wilich
2	2012-06-03	Pardubice	4	3	Hemzel

Rekord: 18 z 79 Bez filtru Wyszukaj

Rys. 1b. Tabela Wyścigi (Microsoft Access)

2. Etapy tworzenia bazy danych

Istotnym etapem tworzenia bazy danych jest określenie celu bazy danych i wymagań, jakie powinna spełniać. Podczas tworzenia profesjonalnych systemów analitycy systemowi zbierają informacje o środowisku, dla którego będą projektowali system informatyczny, m.in. przeprowadzają wywiady, zapoznają się z zasadami funkcjonowania wybranego działu przedsiębiorstwa, analizują charakter i typ produkcji, przeglądają istniejące formularze, raporty i inne dokumenty.



Aby prawidłowo (zgodnie z oczekiwaniami zleceniodawcy) zaprojektować bazę danych, należy wiedzieć nie tylko, jakie dane mają być gromadzone w bazie, ale również, jak mają być przetwarzane i co mamy otrzymać w wyniku ich przetwarzania.

Bazę danych w programie Microsoft Access będziemy tworzyć w następujących etapach:

1. Określenie celu i wymagań bazy danych.
2. Ustalenie zbiorów informacji, które będą gromadzone i przetwarzane w bazie danych.
3. Zaplanowanie nazw i typów pól oraz utworzenie tabel.
4. Zdefiniowanie (niezbędnych do utworzenia relacji) kluczy podstawowych.
5. Definiowanie relacji.
6. Tworzenie formularzy.
7. Wprowadzenie danych.
8. Przygotowanie kwerend.
9. Przygotowanie wymaganych raportów.

3. Na czym polega relacja w bazie danych?

Wiele grup informacji jest ze sobą powiązanych, np. wielu uczniów chodzi do tej samej klasy, jeden człowiek może mieć kilka kont bankowych i kilka adresów e-mail, wielu dostawców może dostarczać ten sam rodzaj owocu do magazynu, a jeden klient może składać wiele zamówień w hurtowni.

W bazie *Stadnina* jeden koń może wziąć udział w wielu wyścigach, a informacja o jednym wyścigu dotyczy tylko jednego konia (co pokazuje tabela na rys. 1b). W projektowanej bazie *Stomatolog* jeden pacjent może przyjść do stomatologa wiele razy, a jedna wizyta będzie dotyczyć tylko jednego pacjenta.

W bazach danych tego rodzaju powiązania powinny zostać odtworzone. W relacyjnej bazie danych określamy je mianem **związków** (w terminologii programu Microsoft Access zwanych **relacjami**).

W programie Microsoft Access dostępne są narzędzia umożliwiające definiowanie związków (relacji) między tabelami w postaci graficznej – jako linii łączącej poszczególne tabele (zob. rys. 2.).



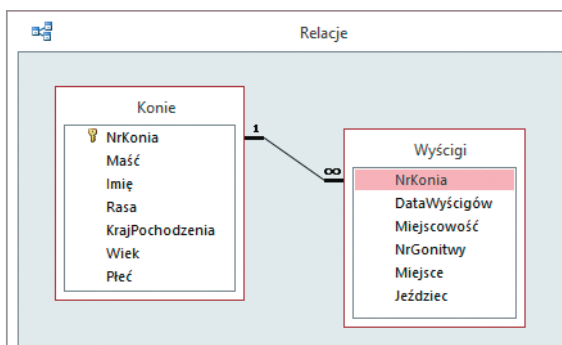
Aby w programie Microsoft Access zdefiniować relację między daną tabelą a inną, jedno pole tej tabeli musi pełnić funkcję **klucza podstawowego**, natomiast w drugiej tabeli musi być wyznaczone pole do zapisywania wartości tego klucza.



Funkcję **klucza podstawowego** może pełnić tylko takie pole, którego wartości są **unikalne** (niepowtarzalne) w ramach tabeli. Klucz podstawowy jednoznacznie identyfikuje rekord.

Wartości pola klucza podstawowego mogą być wprowadzane automatycznie przez program. W tym celu należy zdefiniować pole typu **Autonumerowanie**. Każdy rekord będzie miał wówczas jednoznacznie przydzielony numer – kolejną liczbę naturalną. Wartości klucza podstawowego mogą być również wprowadzane przez użytkownika systemu, ale wówczas typ pola musi być inny niż **Autonumerowanie**.

W tabeli *Konie* funkcję klucza podstawowego pełni pole *NrKonia*, które służy także do definiowania relacji między tabelą *Konie* a tabelą *Wyścigi* (zob. rys. 2.). W tabeli *Wyścigi* wyznaczono pole *NrKonia*, w którym zapisywana jest wartość identyfikatora



Rys. 2. Przykład relacji określonej między dwiema tabelami (Konie i Wyścigi)

danego konia (pobrana z pola *NrKonia* w tabeli *Konie*). Takie pole nazywane jest **kluczem obcym**, ponieważ identyfikuje rekordy z obcej tabeli (tu z tabeli *Konie*).

W naszym przykładzie pola klucza podstawowego i klucza obcego mają taką samą nazwę (*NrKonia*), ale można je nazwać odmiennie.



Aby pole klucza podstawowego i odpowiadające mu pole klucza obcego zostały powiązane relacją, muszą być tego samego typu.

W naszym przykładzie pole *NrKonia* w tabeli *Konie* jest typu **Autonumerowanie**, a więc zawiera liczby. Z tego powodu powiązane relacją pole *NrKonia* w tabeli *Wyścigi* również powinno być typu liczbowego. Nie można jednak nadać polu *NrKonia* w tabeli *Wyścigi* typu **Autonumerowanie**, ponieważ wówczas podczas dopisywania nowych rekordów pola w obydwu tabelach numerowałyby się niezależnie.



Ćwiczenie 2.

1. Na rys. 3a i 3b, pokazano tabele *Konie* i *Wyścigi* w **Widoku projektu**. Wskaż odpowiednio klucz podstawowy i klucz obcy. Sprawdź, jakie typy pól określono dla tych kluczy i wyjaśnij ustnie dlaczego.
2. Zwróć uwagę na wartości obydwu kluczy na rys. 1a i 1b. Odpowiedz ustnie na pytanie: Dlaczego wartości pola *NrKonia* w tabeli *Konie* nie powtarzają się, a w tabeli *Wyścigi* można znaleźć powtarzające się wartości tego pola?

Konie	
Nazwa pola	Typ danych
NrKonia	Autonumerowanie
Maść	Krótki tekst
Imię	Krótki tekst
Rasa	Krótki tekst
KrajPochodzenia	Krótki tekst
Wiek	Liczba
Płeć	Krótki tekst

Rys. 3a. Tabela *Konie* w **Widoku projektu**

Wyścigi	
Nazwa pola	Typ danych
NrKonia	Liczba
DataWyścigów	Data/Godzina
Miejscowość	Krótki tekst
NrGonitwy	Liczba
Miejsce	Liczba
Jeździec	Krótki tekst

Rys. 3b. Tabela *Wyścigi* w **Widoku projektu**

W relacyjnej bazie danych można określić różne typy relacji. W omawianej bazie określono relacje typu „jeden do wielu”, ponieważ jeden koń może wziąć udział w kilku wyścigach, a informacja o jednym wyścigu dotyczy tylko jednego konia. Jest to najczęściej występujący typ relacji.

Inne typy relacji to: „jeden do jednego” i „wiele do wielu”.



Relacja „**jeden do wielu**” występuje, gdy rekord z pierwszej tabeli (tu: *Konie*) powiązany jest z wieloma rekordami drugiej tabeli (tu: *Wyścigi*), natomiast rekord z drugiej tabeli powiązany jest tylko z jednym rekordem z pierwszej.

4. Określenie wymagań i ustalenie zbiorów informacji

Sposób tworzenia bazy danych pokażemy na przykładzie systemu obsługi pacjentów w gabinecie stomatologicznym. Stomatolog gromadzi na kartach, które wypełnia ręcznie, informacje o pacjentach i ich kolejnych wizytach. Naszym zadaniem będzie umożliwienie gromadzenia i przetwarzania tych danych z wykorzystaniem komputera.



Przykład 2. Określenie wstępnych wymagań systemu obsługi pacjentów

Dla bazy *Stomatolog* ustalimy następujące wymagania:

1. Wprowadzanie informacji o pacjentach.
2. Wprowadzanie informacji o wizytach pacjentów.
3. Uzyskiwanie informacji, którzy pacjenci mieli danego dnia wizytę, jakie zabiegi zostały przeprowadzone i jaki był ich koszt (tzw. raport dzienny).
4. Drukowanie odpowiedniego raportu.

Na podstawie wstępnie określonych wymagań dotyczących rodzaju wprowadzanych informacji, uzyskiwanych wyników zapytań do bazy oraz informacji wyprowadzanych w postaci wydruków, możemy ustalić, jakie dane powinny być gromadzone w bazie. Na ich podstawie zaplanujemy tabele.



Przykład 3. Określenie przykładowych zbiorów informacji

Dla bazy *Stomatolog* ustalimy następujące zbiory informacji:

Informacje o pacjencie: nazwisko, imię, ulica i numer domu, kod pocztowy, miejscowość, numer PESEL, numer telefonu.

Informacje o wizycie: data i godzina wizyty, rodzaj wykonanego zabiegu, koszt wizyty.

5. Tabele

5.1. Zasady tworzenia tabel



Do jednej tabeli należy przypisać jedną kategorię informacji.

Korzystając z zaplanowanych dla bazy *Stomatolog* zbiorów informacji, opracujemy dwie tabele: jedną o nazwie *Pacjenci*, służącą do gromadzenia informacji o pacjencie, a drugą o nazwie *Wizyty*, dotyczącą wizyt w gabinecie stomatologicznym.



Należy unikać powtarzania tych samych informacji w kilku tabelach.

Jeśli umieścimy informację o adresie pacjenta w obydwu tabelach, każdą zmianę adresu trzeba będzie nanosić w obydwu miejscach.



W każdej kolumnie powinno się umieszczać pojedynczą informację.

Gdy umieścimy imię i nazwisko pacjenta w oddzielnych kolumnach, ułatwione będzie na przykład porządkowanie tych danych według nazwiska. W przypadku umieszczenia nazwiska i imienia w jednym polu, mogłoby dochodzić do sytuacji, w której użytkownik wpisze najpierw imię, a potem nazwisko.



Nazwa bazy, nazwy tabel i innych obiektów bazy powinny odzwierciedlać ich zawartość. Warto też przyjąć jednolitą dla całej bazy konwencję nazewnictwa tych obiektów.

W programie Microsoft Access bazę zapisujemy pod określoną nazwą w pliku z rozszerzeniem *mdb* lub *mdbx*. Poszczególne obiekty nazywamy oddzielnie, ale zapisywane są one automatycznie w tym samym pliku.



Ćwiczenie 3.

1. Korzystając z rys. 1a (str. 105) i 1b (str. 106) oraz 3a i 3b (str. 108), sprawdź, czy spełnione zostały podane zasady tworzenia tabel.
2. Odpowiedz ustnie na pytania: *Jaką konwencję nazywania obiektów przyjęto w tej bazie? Podaj typy pól występujących w tabelach Konie i Wyścigi.*

5.2. Dlaczego bazy danych tworzy się w kilku tabelach?

Przypuśćmy, że wszystkie informacje w bazie *Stadnina* zapisane są w jednej tabeli. Jeden rekord tej tabeli zawierałby wówczas dane konia i informacje o jego wyścigu. Po każdym wyścigu do tabeli trzeba by było dopisać rekord z wszystkim danymi. Wielokrotnie powtarzałyby się dane konia w każdym rekordzie. Aby uniknąć tej sytuacji, wprowadzono więc dane do dwóch tabel – oddzielnie informacje o koniu i oddzielenie o jego wyścigach.

Baza *Stadnina* z punktu widzenia profesjonalnego tworzenia baz danych powinna składać się z większej liczby tabel powiązanych relacjami (uproszczono ją w celach dydaktycznych). W tabelach powtarzają się bowiem informacje o miejscowościach, rasach koni i ich maściach. Te informacje również powinny być zapisane w oddzielnych tabelach.

5.3. Tworzenie tabel

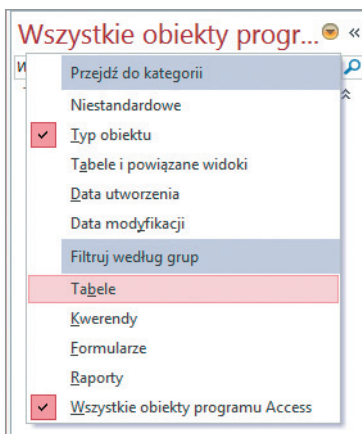
W kolejnym etapie zaplanujemy odpowiednie pola, aby spełnić wymagania określone w przykładzie 2. i móc zapisywać informacje określone w przykładzie 3.



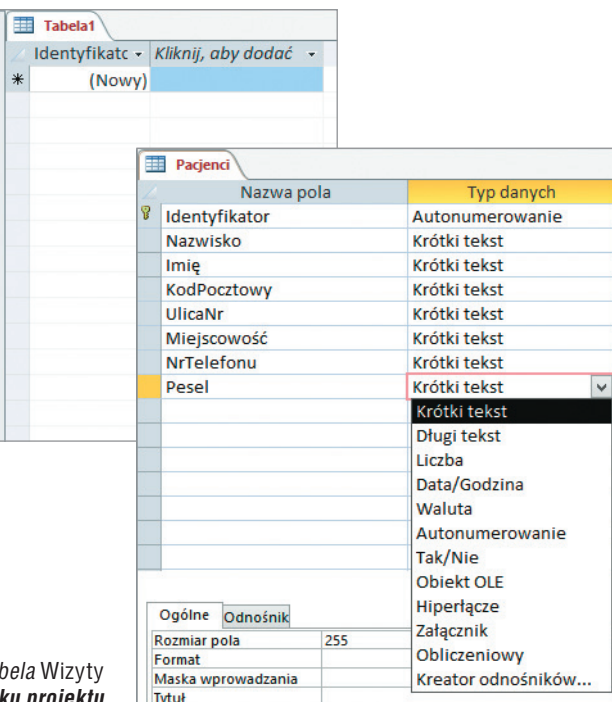
Przykład 4. Tworzenie tabeli *Pacjenci* w Widoku projektu

Chcemy utworzyć nową bazę danych, a w niej tabelę o nazwie *Pacjenci*.

1. Utwórz nową (pustą) bazę danych i z panelu wyboru obiektów (widocznego z lewej strony okna programu) wybierz opcję tworzenia tabeli (polecenie **Tabele**) – rys. 4.
2. Nazwij i zapisz bazę w pliku oraz nazwij nową tabelę, gdy program tego zażąda.
3. Wybierz tworzenie tabeli w **Widoku projektu** (opcja **Utwórz tabelę w widoku projektu** lub **Widok** na karcie **Plik**).



Rys. 4. Tabela Pacjenci
w Widoku projektu



Rys. 5. Tabela Wizyty
w Widoku projektu

4. Zdefiniuj pole klucza podstawowego typu **Autonumerowanie** (jeśli program nie wstawił domyślnie pola klucza o nazwie *Identyfikator*). Przy nazwie pola klucza powinien być umieszczony kluczyk (zob. rys. 5.). Jeśli nie został on wstawiony automatycznie, można go wstawić, klikając na pasku narzędzi programu

ikonę klucza podstawowego ().

W tabeli *Pacjenci* klucz podstawowy jest potrzebny do jednoznacznej identyfikacji rekordów oraz do zdefiniowania relacji między tą tabelą a tabelą *Wizyty*.

5. Wpisz pozostałe nazwy pól i określ ich typy, wybierając je z listy wyboru (zob. rys. 5.).

Uwaga:

- W starszych wersjach programu podczas próby zapisania tabeli, w której nie zdefiniowano klucza podstawowego, program zleci jego zdefiniowanie.
- Widok tabeli (np. **Widok projektu**, **Widok arkusza danych**) można zmienić, korzystając z menu kontekstowego, wybieranego z obszaru paska tytułu tabeli,

lub klikając odpowiedni przycisk (np.  , ).



Ćwiczenie 4.

1. Utwórz nowy plik bazy danych. Zapisz plik bazy danych pod nazwą *Stomatolog*.
2. Korzystając z przykładu 4., utwórz nową tabelę.
3. Wprowadź nazwy pól, korzystając z rys. 5. Zmień nazwę klucza podstawowego *Identyfikator* na nazwę *NrPacjenta*. Dla każdego z pozostałych pól pozostaw domyślny, tekstowy typ danych. Zapisz tabelę, zamykając okno tabeli.

W bazie *Stomatolog* utworzymy drugą tabelę (*Wizyty*), w której, poza informacjami o wizycie, wprowadzimy pole o nazwie *NrPacjenta* typu **Liczba**. Pole to pełnić będzie funkcję klucza obcego, podobnie jak pole *NrKonia* w tabeli *Wyścigi* w bazie *Stadnina*.



Ćwiczenie 5.

1. Utwórz tabelę o nazwie *Wizyty*. Wprowadź nazwy pól i ich typy, korzystając z rys. 6.
2. Zapisz tabelę. Nie definiuj w tej tabeli klucza podstawowego. Jeśli klucz został zdefiniowany automatycznie – usuń klucz, zmień nazwę pola na *NrPacjenta*, a typ danych – na **Liczba**.
3. Dlaczego pole *NrPacjenta* nie może być typu **Autonumerowanie**?

Wskazówki:

- Aby usunąć klucz, kliknij wybrane pole, a następnie przycisk z kluczykiem.
- W nowszych wersjach programu opcję dodawania nowej tabeli znajdziesz na karcie **Tworzenie**.



Nazwa pola	Typ danych
NrPacjenta	Liczba
DataWizyty	Data/Godzina
GodzinaWizyty	Data/Godzina
RodzajWizyty	Krótki tekst
KosztWizyty	Liczba

Rys. 6. Tabela *Wizyty* w *Widoku projektu*

5.4. Definiowanie relacji



Ćwiczenie 6.

Korzystając z informacji opisanych w punkcie 3., odpowiedz na pytania:

1. Czy zdefiniowanie klucza podstawowego w tabeli *Pacjenci* wystarczy, aby określić relację między tabelami *Pacjenci* i *Wizyty*? Odpowiedź uzasadnij.
2. Dlaczego w bazie *Stomatolog* należy zdefiniować relację typu „jeden do wielu”?

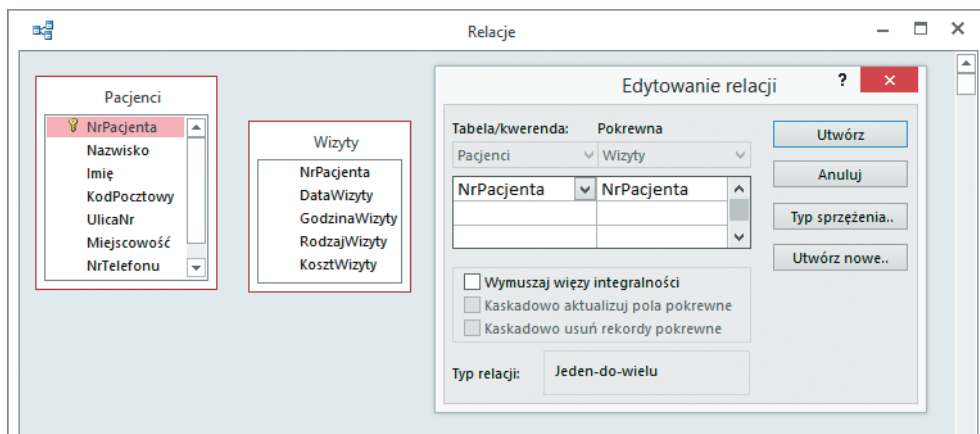


Przykład 5. Definiowanie relacji między tabelami

Chcemy zdefiniować relację między tabelami *Pacjenci* i *Wizyty*.

1. Wybierz polecenie **Relacje** (w menu **Narzędzia** lub na karcie **Narzędzia bazy danych**).
2. Wybierz tabele, między którymi ma być określona relacja (tu: *Pacjenci* i *Wizyty*). Aby umieścić tabele w oknie **Relacje**, należy kliknąć dowolne miejsce w obszarze okna i z menu kontekstowego wybrać polecenie **Pokaż tabelę**.
3. Przeciągnij pole klucza podstawowego na odpowiadające mu pole klucza obcego w drugiej tabeli – otworzy się okno **Edytowanie relacji** (zob. rys. 7.).
4. Zaznacz opcję **Wymuszaj więzy integralności** i kliknij przycisk **Utwórz**.

Uwaga: Wymuszanie więzów integralności oznacza, że nie ma możliwości wprowadzenia do pola klucza obcego wartości nieistniejących w polu klucza podstawowego, czyli w tabeli *Wizyty* nie można dopisać rekordu dla pacjenta, którego nie ma w tabeli *Pacjenci*.



Rys. 7. Okno **Edytowanie relacji** w widoku **Relacji**



Ćwiczenie 7.

1. Korzystając z przykładu 5, zdefiniuj w bazie *Stomatolog* relację między tabelami *Pacjenci* i *Wizyty*.
2. Zapisz plik pod tą samą nazwą.



Warto zapamiętać

- Baza danych to zbiór danych, czyli odpowiednio zorganizowanych informacji, które można przetwarzać za pomocą programów komputerowych.
- Wykonywanie operacji na danych, czyli przetwarzanie, to m.in. ich wprowadzanie, redagowanie, wyszukiwanie, prezentowanie.
- Dane w relacyjnych bazach danych przechowujemy w tabelach. Tabele składają się z wierszy (rekordów) i kolumn (pól). W obrębie tabeli nazwy pól nie mogą się powtarzać.
- W programie Microsoft Access definiuje się klucz podstawowy, który jednoznacznie identyfikuje rekordy w bazie.
- Aby w programie Microsoft Access zdefiniować relację między dwiema tabelami, jedno pole pierwszej tabeli musi pełnić funkcję klucza podstawowego, natomiast w drugiej tabeli musi być wyznaczone pole do zapisywania wartości tego klucza. Pola tych kluczy muszą być tego samego typu.
- Nie należy powtarzać tych samych informacji w kilku tabelach.



Pytania i polecenia

1. Czym jest baza danych?
2. Na czym polega przetwarzanie danych w bazie danych?
3. Jak są zorganizowane dane w relacyjnej bazie danych?
4. Czym jest rekord? Podaj przykład.
5. Podaj etapy przygotowania w programie Microsoft Access prostej relacyjnej bazy danych.

6. Dlaczego istotne jest wstępne określenie wymagań dotyczących rodzaju wprowadzanych informacji do bazy danych?
7. Czym jest relacja?
8. Jakie warunki muszą być spełnione, aby zdefiniować relację między dwiema tabelami?
9. Jaką funkcję pełni klucz podstawowy, a jaką klucz obcy?
10. Dlaczego pole *NrKonia* (w bazie *Stadnina*) jednoznacznie identyfikuje rekordy w tabeli *Konie*?
11. Kiedy występuje relacja typu „jeden do wielu”?
12. Omów na przykładach zasady tworzenia tabel.



Zadania

1. Korzystając z przykładów i rysunków zamieszczonych w tym temacie, utwórz bazę danych *Stadnina*: przygotuj dwie tabele *Konie* i *Wyścigi* oraz zdefiniuj odpowiednią relację między nimi.
2. W magazynie dżemów klienci składają zamówienia na różne rodzaje dżemów. Jeśli klient chce zamówić kilka rodzajów dżemów, na każdy z nich składa oddzielne zamówienie. Przygotuj bazę danych składającą się z dwóch tabel: *Klienci* i *Zamówienia*. Tabela *Klienci* powinna zawierać m.in.: nazwisko, imię i adres klienta, a tabela *Zamówienia* informacje o zamówieniu, np. nr zamówienia, rodzaj dżemu, liczbę sztuk, datę złożenia zamówienia, datę realizacji. Zdefiniuj relację między tabelami, odwzorowując rzeczywiste powiązania. Zapisz bazę w pliku pod nazwą *Magazyn dżemów*.
Uwaga: Profesjonalny system obsługi magazynu zawiera więcej powiązanych tabel i danych.
3. Przygotuj bazę *Zawody sportowe*, składającą się z dwóch tabel: *Zawodnicy* (do przechowywania danych o zawodnikach) i *Zawody* (do zapisywania informacji o zawodach, w których uczestniczy dany zawodnik). Określ odpowiednie zbiory informacji, które będą przechowywane w obydwu tabelach. Następnie ustal odpowiednio pola i określ ich typy. Zdefiniuj relacje, biorąc pod uwagę, że jeden zawodnik może wziąć udział w kilku zawodach, a informacje o jednych zawodach dotyczą konkretnego zawodnika.
4. Zaprojektuj bazę danych: ustal zbiory informacji, zastanów się nad rzeczywistymi powiązaniami między grupami informacji, określ wymagania bazy, zaplanuj pola i tabele. Zdefiniuj odpowiednie klucze i relacje.
 - a. *Wypożyczalnia samochodów* (np. informacje o samochodach, klientach, transakcjach wypożyczenia).
 - b. *Moja płytkoteka* (np. informacje o albumach, wykonawcach, piosenkach).
 - c. *Obsługa schroniska* (np. informacje o pokojach, gościach, rezerwacjach).
 - d. *Serwis sprzętu komputerowego* (np. informacje o klientach, sprzęcie, reklamacjach).

Dla zainteresowanych

5. Zaprojektuj bazę danych *Klinika zwierząt*. Ustal zbiory informacji i wymagania systemu, zastanów się nad rzeczywistymi powiązaniami między grupami informacji, zaplanuj pola i tabele. Zdefiniuj odpowiednie klucze i relacje. Baza powinna składać się z trzech tabel powiązanych relacjami (*Właściciele*, *Zwierzaki*, *Wizyty*). Zakładamy, że jeden właściciel może mieć kilka zwierząt, z którymi przyjdzie na wizytę tego samego dnia, ale informacja o wizycie zapisywana jest oddzielnie dla każdego ze zwierząt.