

LEKCJA 1 Bazy danych

Jednym z najważniejszych obecnie zadań komputerów jest gromadzenie, przechowywanie i udostępnianie użytkownikom danych. Informacje przechowywane są zazwyczaj w sposób uporządkowany, co ułatwia ich wyszukiwanie, z tego względu przechowuje się je w uporządkowanych strukturach zwanych **bazami danych**.

Prostymi przykładami baz danych są np. dziennik lekcyjny zawierający spis uczniów, inne dane o uczniu, otrzymywane oceny i informacje o obecności. Dobrym przykładem również biblioteka, wypożyczalnia kaset itp. W bibliotece czytelnik korzysta przede wszystkim z katalogu książek, który zawiera informacje o wszystkich książkach, jest też spis książek wypożyczonych (zawierający informację o osobie, która wypożyczyła książkę) i jest również rejestr osób, które mogą korzystać z zasobów biblioteki (zawierający przy każdej z nich spis wypożyczonych książek). Informacje umieszczone w poszczególnych spisach są ze sobą powiązane.

Zbiory baz danych muszą być poddawane różnym czynnościom z ich tworzeniem i utrzymaniem. Żeby inni użytkownicy mieli możliwość korzystania z informacji należy oczywiście wprowadzić je do komputera i umożliwić dostęp do nich określonej grupie pracowników. Utrzymywanie baz wiąże się z koniecznością ciągłego aktualizowania (uzupełnianie i korygowanie) a także zabezpieczania przed utratą danych. Książki telefoniczne aktualizuje się zazwyczaj, co roku. Ale np. informacja o rezerwacji biletów lotniczych musi być aktualizowana w ciągu kilku sekund w tysiącach biur podróży na całym świecie i jest praktycznie niemożliwe by odbywało się to bez pomocy komputera.

W języku potocznym określenie **baza danych** jest używane zarówno na określenie **zbioru informacji** w odpowiedniej strukturze, jak i **programu obsługi baz danych (bazodanowy)**. Programem, który wyznaczył przed laty swoisty standard w zakresie komputerów osobistych jest **dBase**. W środowisku Windows takim standardem jest niewątpliwie MS Access.

Budowa komputerowych baz danych

Zestaw informacji dotyczących jednego elementu komputerowej bazy danych jest nazywany **rekordem**. Z reguły rekord zawiera informacje powiązane ze sobą (np. informacje o książkach, informacje o czytelnikach itp.). Rekord ma określoną strukturę: składa się z pewnej liczby **pól** zapisanych w określonej kolejności. Każde pole rekordu ma określoną **nazwę, długość i typ**. Typy pól określają rodzaj danych, jaki może być umieszczony w polu (liczbowe, tekstowe, daty, logiczne).

(Przy określaniu typów pól należy się kierować możliwościami wykonywania na ich zawartości działań. Na danych znakowych można wykonywać operacje porównań. Na danych numerycznych możemy dokonywać operacji arytmetycznych i porównań. W przypadku dat można nie tylko sprawdzać ich kolejność, ale także obliczać odległość między nimi.. Na danych logicznych można dokonać tylko operacji logicznych. Na danych notatnikowych nie dokonuje się żadnych operacji, poddaje się je jedynie edycji).

Dane gromadzi się w **tabelach**, w których poszczególne wiersze są rekordami bazy, a każda kolumna zawiera dane określonego typu – pola. Aby tabela spełniała wymagania bazodanowe, pierwszy jej wiersz powinien zawierać opisy poszczególnych kolumn-pól.

Projektowanie bazy danych

Załóżmy, że zarządzamy domowym zbiorem książek, który zawiera kilkadziesiąt pozycji. Książki te (niestety) lubimy pożyczać i odnotowujemy to w komputerowej tabeli. Tabela z wpisami wygląda w następujący sposób:

AUTOR	TYTUŁ	WYPOŻYCZYŁ	ADRES	DATA	ZWROT
Mickiewicz	Dziady	Libront Waclaw	Bobowa	2000-01-01	2000-01-31
Mickiewicz	Pan Tadeusz	Libront Waclaw	Bobowa	2000-01-01	2000-01-31
...					

Jest to przykład tzw. **tabeli jednorodnej**, w której każdy wiersz zawiera wszystkie niezbędne informacje. Ręczne prowadzenie takiej tabeli wymaga nieustannego przepisywania tych samych informacji (tytuły książek, nazwiska pożyczających) – dużo pracy i możliwość pomyłek. A wyobraźmy sobie, że w ciągu całego roku uzbierało się ponad tysiąc wpisów, a my chcemy znaleźć, kto nam jeszcze nie oddał książki?!

(zobacz przykładową tabelę w Excel: DOMOWA WYPOŻYCZALNIA.XLS)

Relacyjne bazy danych

Wszystkie te niedogodności można zlikwidować, jeśli taką dużą tabelę podzielimy na mniejsze table, które będą zawierać dane na określony temat:

KSIĄŻKI – informacje o książkach

NRKS	AUTOR	TYTUŁ	WYDAWNICTWO	STRON	CENA	TYP
------	-------	-------	-------------	-------	------	-----

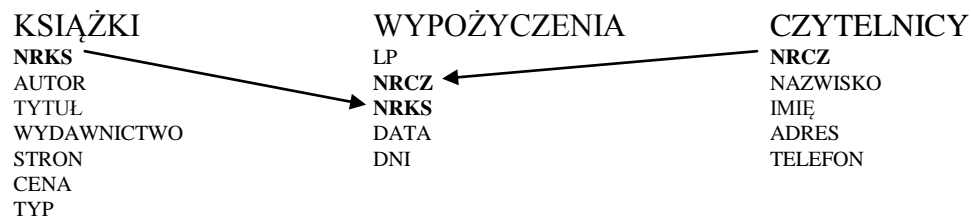
CZYTELNICZY – dane czytelników

NRCZ.	NAZWISKO	IMIĘ	ADRES	TELEFON
-------	----------	------	-------	---------

WYPOŻYCZENIA (kto, co, kiedy i na jak długo wypożyczył)

LP	NRCZ	NRKS	DATA	ILE
----	------	------	------	-----

Tabele KSIĄŻKI i CZYTELNICZY zawierają tylko **po jednym** wpisie na każdą książkę i każdego czytelnika – unikamy w ten sposób **nadmiarowości** (informacje się nie powtarzają). Dodatkowo każdy wiersz tabeli oznaczony jest specjalnym **kluczem (NACZ i NRKS)**, który pozwala je rozróżniać. Tabela wypożyczenia zawiera jedynie **numer** książki (NRKS) i **numer** wypożyczającego (NRCZ.), a nie całą informację (zmniejszamy wielkość tabeli). Te numery wiążą ze sobą trzy table określonymi zależnościami (powiązaniem). Powiązania te w bazach danych nazywamy **relacjami**.



Tak skonstruowany zestaw tabel wraz z zależnościami (powiązaniem) między nimi nosi nazwę **relacyjna baza danych**. Struktura bazy danych się komplikuje w porównaniu do jednej tabelki, ale zajmuje mniej miejsca w pamięci i jest bardziej przejrzysta. Od poprawnego skonstruowania tabel i zależności zachodzących między nimi zależy działanie bazy (szybkość i wygoda uzyskiwania informacji).

(zobacz przykładową relacyjną bazę w Access: KSIĘGARNIA.MDB)

(Żeby zdać sobie sprawę z korzyści wynikających z redukcji powtarzających się danych, wyobraźmy sobie, że w ciągu roku wypożyczyliśmy jednej osobie 100 książek. Zamiast 100 razy pisać nazwisko, imię, adres itp., wpisujemy 100 razy tylko jeden i ten sam numer)

Problemy obsługi baz danych

Problem zbyt wielu danych i przejrzystość są wystarczającymi powodami, aby przekonać projektanta bazy do stosowania relacyjnych baz danych, a unikania tabel jednorodnych. Jednak stosowanie relacyjnych baz danych również wiąże się z problemami. Są to tzw. **anomalie**.

anomalie wstawiania - te same zabawki ale różne ceny, użytkownik wybiera raz jedną, raz drugą lub problem ze znalezieniem opisu zabawki, która jeszcze nie była sprzedawana

anomalie modyfikacji - jeśli dane wszystkich zabawek były w jednej tabeli, to po zmianie ceny jednej trzeba zmienić cenę wszędzie tam, gdzie występowała ta zabawka

anomalie usuwania - usuwamy z tabeli zapis o zabawce, ale także jej cenę, jak potem policzyć np. wartość sprzedaży gdy nie ma już ceny zabawki

spójność bazy - czy wprowadzający dane nie robi błędów

poufność danych - do pewnych danych powinni mieć dostęp tylko wybrani użytkownicy bazy danych i powinni mieć możliwość wykonywania na tych danych tylko wybranych operacji (wstawianie nowych, usuwanie, modyfikowanie itp.)

współbieżność - jednoczesny dostęp do danych kilku użytkowników jeśli pracujemy w sieci. Tylko jeden użytkownik może modyfikować rekordy, podczas gdy reszta może tylko je oglądać. Jak przydzielić, który ma być pierwszy, jaka powinna być kolejność dopuszczania użytkowników do możliwości modyfikacji, żeby zależności w bazie prawidłowo się przeliczały (np. ktoś wprowadza nowe dane, zmienił cenę zabawki i nie zdążył jeszcze wpisać nowej ilości, a w tym czasie inny użytkownik już sprzedaje zabawki po nowej cenie, których jeszcze nie ma w bazie)

zabezpieczenie, archiwizowanie danych - jak często kopiować duże zbiory i gdzie przechowywać kopie zabezpieczające. Gdy zginie komputer nie ma problemu - można kupić nowy. Gdy zginą dane - firma nie może pracować.

Manipulowanie danymi

Przekształcanie danych zawartych w bazach danych, tak by można było uzyskać z nich potrzebne informacje, opiera się na trzech podstawowych operacjach: **selekcji**, **projekcji** i **połączenia**. Za pomocą projekcji usuwamy wybrane kolumny z tabeli. Selekcja usuwa wybrane rekordy z tabelki (z reguły odbywa się to za pomocą wyszukiwania rekordów spełniających pewne warunki logiczne - **filtry**). Połączenie polega na łączeniu tabel według określonych kryteriów (kluczy).

Programy obsługi baz danych mogą również porządkować dane dwoma sposobami. **Sortowanie** przypomina operację układania kart podczas gry. Po otrzymaniu kart w przypadkowej kolejności, gracz układa je np. kolorami. Podczas sortowania zmienia się kolejność rekordów w bazie, z reguły tworzona jest wtedy nowa baza zawierająca uporządkowane rekordy. Sortowanie nie rozwiązuje problemu, gdy posługujemy się tabelą, którą chcemy mieć uporządkowaną raz np. według nazwisk, a za chwilę według dat urodzenia. Zajmuje to dużo czasu i miejsca na dysku, z tego względu stosuje się także **indeksowanie**. Efekt indeksowania jest podobny, ale mechanizm inny. Program nie zmienia kolejności rekordów w tabeli, ale tworzy pomocniczą tabelę informującą o kolejności rekordów, jaka istniałaby po sortowaniu. Baza danych pozostaje nienaruszona.

Informacja w bazie powinna być czasem przedstawiona w postaci **raportu**, który z reguły zawiera: informacje o rekordach, zbiorcze informacje dotyczące grup rekordów spełniających określone kryteria, informacje pomocnicze. Raporty mogą być wyświetlane na ekranie lub drukowane.

Przy częstym posługiwaniu się bazą danych warto dopracować układ i postać graficzną przedstawianych na ekranie informacji. Tabela jest czasem mało przejrzysta i można ją zastąpić graficznie dopracowanym **formularzem**. Tworzy się formularze do wyświetlania danych na ekranie do modyfikowania rekordów i wpisywania nowych. W wielu przypadkach programy do obsługi baz danych oferują tzw. **wzorce**, za pomocą których można szybko utworzyć efektowny formularz.

Obiekty bazy danych

TABELA – podstawowa struktura służąca do przechowywania danych. Powinna być tak zaprojektowana, aby były to informacje dotyczące jednego tematu, co eliminuje duplikowanie się danych. Każdy wiersz tabeli – **rekord** – zawiera dane dotyczące jednego elementu tabeli (np. informacje o czytelniku), a każda kolumna tabeli – **pole** – zawiera dane jednego typu (np. nazwisko czytelnika).

KWERENDA - służy do wyświetlania, zmiany i analizy danych zawartych w tabelach. Wynikiem działania kwerendy jest tabela (z reguły wirtualna).

FORMULARZ – wyświetlanie danych zawartych w tabelach w odpowiedni sposób, dogodny dla użytkownika.

RAPORT – głównym zastosowaniem jest odpowiednie przygotowanie danych do wydruku

Projektowanie tabel

Konstruując tabele powinniśmy wzorować się na zależnościach występujących w otaczającym nas świecie. Zarówno tabele, jak i łączące je relacje powinny być w ten naturalny sposób odwzorowane w bazie. Aby nie dochodziło w systemie bazodanowym do różnorodnych anomalii, należy podczas konstruowania tabel stosować się do poniższych wskazówek:

- każda tabela powinna zawierać informacje tylko na jeden temat
- w każdej kolumnie powinna być umieszczona tylko pojedyncza informacja
- tabele i pola powinny być jednoznacznie ponazywane
- nie powtarzamy tych samych informacji w różnych tabelach
- w jednym polu nie przechowujemy listy danych

ĆWICZENIA

Załącz bazę danych o nazwie „Wypożyczalnia”

Utwórz trzy tabele: Książki, Czytelnicy, Wypożyczenia – kolumny opisane w poniższych tabelach. Kolumna „Inne” zawiera wymagania dotyczące właściwości pól. Zmień tylko te, które są opisane. Innym pozostaw ustawienia domyślne. Połącz tabele relacjami. Wprowadź dowolne dane.

Książki

Nazwa	Typ	Inne
NRKS	Autonumerowanie	klucz podstawowy
TYTUL	Tekst	rozmiar - 150, wymagane-TAK
AUTOR	Tekst	rozmiar – 50
WYDAWNICTWO	Tekst	rozmiar – 50
STRON	Liczba	wymagane-TAK, rozmiar - liczba całkowita
CENA	Liczba	rozmiar – pojedyncza precyzja, format – walutowy
TYP	Tekst	wymagane-TAK, wartość domyślna: „biuro”

Studenci

Nazwa	Typ	Inne
NRCZ	Autonumerowanie	klucz podstawowy
NAZWISKO	Tekst	rozmiar – 20, wymagane-TAK
IMIE	Tekst	rozmiar – 30, wymagane-TAK
ADRES	Tekst	rozmiar – 100, wymagane-TAK
TELEFON	Tekst	maska – numer telefonu, wymagane-TAK

Wypożyczenia

Nazwa	Typ	Inne
LP	Autonumerowanie	klucz podstawowy
NRCZ	Liczba	wymagane-TAK
NRKS	Liczba	wymagane-TAK
DATA	Data	wymagane-TAK
DNI	Liczba	

MASKA

Maski wprowadzania określają sposób wprowadzania przez użytkowników danych do bazy danych. Przy użyciu maski wprowadzania można na przykład wymusić na użytkownikach wprowadzanie informacji w określonej postaci, np. numeru telefonu w Polsce +48 00 000-00-00.

Maski wprowadzania mogą ułatwić zapobieganie wprowadzaniu przez użytkowników nieprawidłowych danych. Dodatkowo zapewniają one spójny sposób wprowadzania danych przez użytkowników, co z kolei ułatwia wyszukiwanie danych i obsługę bazy danych.

PRZYKŁADY

Definicja maski	Przykłady	Opis
\(900\) 00-00-000 lub \ (900\) "00-00-000	(018)35-14-009 (18)35-14-009	w nawiasach dwie lub trzy cyfry numeru kierunkowego (pierwsza niewymagana) i dokładnie 7 cyfr przedzielonych pauzami
#999	-20 2000	maksymalnie 4 cyfry lub maksymalnie trzy cyfry poprzedzone znakiem plus lub minus
>L<????????????	Wacław Libront	Pierwsza litera wymagana i zawsze duża, pozostałe 14 małe
>L????L?000L0	GRENGR339M 3 MAY R 452B7	wszystkie litery duże, pierwsza litera wymagana, kolejne 4 niewymagane, kolejna wymagana, niewymagana, trzy wymagane cyfry, litera i cyfra
>L0L\ 0L0	T2F 8M4	duże litery, wszystkie litery i cyfry wymagane, pomiędzy nimi odstęp
"ISBN "0-&&&&&&-0	ISBN 1-15-507-7 ISBN 0-1-3962-5	napis ISBN, odstęp, wymagana cyfra, pauza, sześć dowolnych znaków, pauza i wymagana cyfra

Symbol	Opis
0	Cyfra (od 0 do 9, pozycja wymagana; znaki plus [+] i minus [-] nie są dozwolone).
9	Cyfra lub spacja (pozycja opcjonalna; znaki plus i minus nie są dozwolone).
#	Cyfra lub spacja (pozycja opcjonalna; puste miejsca konwertowane na spacje, znaki plus i minus dozwolone).
L	Litera (od A do Z, pozycja wymagana).
?	Litera (od A do Z, pozycja opcjonalna).
A	Litera lub cyfra (pozycja wymagana).
a	Litera lub cyfra (pozycja opcjonalna).
&	Dowolny znak lub spacja (pozycja wymagana).
C	Dowolny znak lub spacja (pozycja opcjonalna).
.,:;- /	Dziesiętny symbol zastępczy oraz separatorzy tysięcy, daty i godziny. (Znak, który faktycznie zostanie użyty zależy od ustawień regionalnych określonych w Panelu sterowania systemu Microsoft Windows).
<	Powoduje, że wszystkie następujące po nim znaki są konwertowane na małe znaki.
>	Powoduje, że wszystkie następujące po nim znaki są konwertowane na duże znaki.
!	Powoduje, że znaki w masce wprowadzania są wyświetlane od strony prawej do lewej zamiast od lewej do prawej. Znaki wpisywane w masce wprowadzania zawsze wypełniają ją od strony lewej do prawej. Znaku wykrzyknika można użyć w dowolnym miejscu w masce wprowadzania.
\	Powoduje, że kolejny znak jest wyświetlany jako znak literalowy. Jest on używany w celu wyświetlenia dowolnego znaku uwzględnionego w tej tabeli jako znak literalowy (na przykład, ciąg „\A” jest wyświetlany jako „A”).