

Relacyjna baza danych

Baza danych to zbiór danych zapisanych w ściśle określony sposób w strukturach odpowiadających założonemu modelowi danych.

1. Wyświetlane relacje mają postać tabeli:

- Nagłówki tabeli są atrybutami=polami=cechami.
- K-wartości=krotki=rekordy są wierszami tabeli.
- Baza danych jest zbiorem relacji.
- Schemat relacji jest zbiorem {atrybut, dziedzina, [ograniczenia integralnościowe]}.
- Schemat bazy danych jest zbiorem schematów relacji.

2. W bazie danych definiuje się związki pomiędzy danymi znajdującymi się w różnych tabelach (związki =relacje). Relacja posiada następujące cechy:

- każdy atrybut relacji ma unikalną nazwę,
- porządek atrybutów w relacji nie jest istotny,
- porządek rekordów w relacji nie jest istotny i nie jest elementem definicji relacji,
- wartości atrybutów są elementarne,
- relacja nie zawiera rekordów powtarzających się. Ponieważ relacja jest zbiorem k-wartości, więc, z definicji zbioru, wszystkie rekordy relacji muszą być unikalne.
- każda tabela musi mieć przypisaną nazwę i nazwa ta powinna określać rodzaj danych, jaki planujemy w tej tabeli przechowywać.

3. W tabeli nie mogą występować dwa identyczne wiersze - unikalność wiersza zapewniona poprzez zdefiniowanie klucza podstawowego **PK** (np. nr pesel, numer albumu studenta, numer dowodu osobistego).

4. Klucz obcy **FK** relacji jest atrybutem lub zbiorem atrybutów, który odwołuje się do klucza podstawowego w innej tabeli.

5. Jak tworzyć tabelę i opis:

Nazwa tabeli
NTA_ PK nazwa klucza podstawowego: typ danych (rozmiar)
NTA_ Atrybut : typ danych (rozmiar)

NTA – „nazwa tabeli”-unikalna- umieszczana przed kluczem i każdym atrybutem.

Przykład z kluczem podstawowym (głównym)

Wydział
WYD_PKid: INTEGER
WYD_Nazwa: VARCHAR(45)
WYD_Skrot: VARCHAR(20)

Opis: Tabela **Wydział** zawiera trzy pola: identyfikator, nazwę i skrót wydziału. Skrót WYD identyfikuje tabelę: Wydział.

Przykład z kluczem podstawowym (głównym) i obcym.

Katedra
KAT_PKid: INTEGER
WYD_FKWydział_id: INTEGER
KAT_Nazwa: VARCHAR(45)
KAT_Skrot: VARCHAR(20)

Opis: Tabela **katedra** zawiera informacje: identyfikator katedry, nazwę wydziału (klucz obcy), nazwę katedry i skrót katedry. Skrót KAT identyfikuje tabelę Katedra. WYD_FK identyfikuje klucz obcy: Wydziału.

Przykład z kluczem podstawowym (głównym) i dwoma kluczami obcymi.

Kierunek
KIE_PKid: INTEGER
WYD_FKWydział_id: INTEGER
RST_FKRodzaj_studiow_id: INTEGER
KIE_Nazwa: VARCHAR(45)
KIE_Skrot: VARCHAR(20)
KIE_Sortowanie: INTEGER

Opis: Tabela **kierunek** zawiera pola: identyfikator kierunku, nazwę wydziału, rodzaj studiów, nazwę kierunku, skrót kierunku oraz wagę stosowaną do sortowania kierunków. Skrót KIE identyfikuje tabelę Kierunek. RST_FK identyfikuje klucz obcy: Rodzaj studiów.

6. Typy danych

Poniżej przedstawiono najpopularniejsze typy danych, rozmiar danych oraz krótki opis.

Typ danych	Rozmiar	Dopuszczalne wartości
CHAR(długość)	długość bajtów	Przechowuje łańcuchy znakowe o długości od 0 do 255 znaków. CHAR jest polem o stałej długości.
VARCHAR(długość)	długość bajtów + 1 bajt	Przechowuje łańcuchy znakowe o zmiennej długości. Długość zmienia się w zależności od ilości znaków do przechowania.
TINYTEXT	długość łańcucha + 1 bajt	Przechowuje łańcuchy znakowe o długości do 255 znaków.
TEXT	długość łańcucha + 2 bajty	Przechowuje łańcuchy znakowe o długości do 65 535 znaków.
MEDIUMTEXT	długość łańcucha + 3 bajty	Przechowuje łańcuchy znakowe o długości do 16 777 215 znaków.

LONGTEXT	długość łańcucha + 4 bajty	Przechowuje łańcuchy znakowe o długości do 4 294 967 295 znaków.
TINYBLOB	długość łańcucha + 1 bajt	Przechowuje łańcuchy binarne do 255 bajtów.
BLOB	długość łańcucha + 2 bajty	Przechowuje łańcuchy binarne o długości do 65 535 bajtów.
MEDIUMBLOB	długość łańcucha + 3 bajty	Przechowuje łańcuchy binarne o długości do 16 777 215 bajtów.
LOB	długość łańcucha + 4 bajty	Przechowuje łańcuchy binarne o długości do 4 294 967 295 znaków.
ENUM('wartość', 'wartość2', ...)	1 lub 2 bajty	Kolumna przechowuje jedną z podanych wartości (maksymalna liczba wartości to 65 535).
SET('wartość', 'wartość2', ...)	1, 2, 3, 4 lub 8 bajtów	Kolumna przechowuje jedną lub więcej z podanych wartości (maksymalna liczba wartości to 64).
TINYINT	1 bajt	Kolumna przechowuje wartości od -128 do 127 lub od 0 do 255 (w zależności od tego, czy posiada atrybut UNSIGNED).
SMALLINT	2 bajty	Kolumna przechowuje wartości od -32768 do 32767 lub od 0 do 65535 (w zależności od tego, czy posiada atrybut UNSIGNED).
MEDIUMINT	3 bajty	Kolumna przechowuje wartości od -8388608 do 8388607 lub od 0 do 16777215 (w zależności od tego, czy posiada atrybut UNSIGNED).
INT	4 bajty	Kolumna przechowuje wartości od -2147483648 do 2147483647 lub od 0 do 4294967295 (w zależności od tego, czy posiada atrybut UNSIGNED).
BIGINT	8 bajtów	Kolumna przechowuje wartości od -9223372036854775808 do 9223372036854775807 lub od 0 do 18446744073709551615 (w zależności od tego, czy posiada atrybut UNSIGNED).
FLOAT(długość, precyzja)	4 bajty	Liczba zmiennoprzecinkowa o maksymalnej liczbie cyfr <i>długość</i> w tym ilość liczb po przecinku (<i>precyzja</i>). FLOAT służy do przechowywania mniejszych wartości, niż w DOUBLE.
DOUBLE(długość, precyzja)	8 bajtów	Liczba zmiennoprzecinkowa o maksymalnej liczbie cyfr <i>długość</i> w tym ilość liczb po przecinku (<i>precyzja</i>).
DATE	3 bajty	Data w formacie YYYY-MM-DD, gdzie YYYY to rok, MM to miesiąc, DD to dzień.
DATETIME	8 bajtów	Data w formacie YYYY-MM-DD HH:MM:SS, gdzie YYYY to rok, MM to miesiąc, DD to dzień, HH to godzina, MM minuty oraz SS to sekundy.
TIMESTAMP	4 bajty	Data w formacie YYYY-MM-DD HH:MM:SS, gdzie YYYY to rok, MM to miesiąc, DD to dzień, HH to godzina, MM minuty oraz SS to sekundy.
TIME	3 bajty	Czas w formacie HH:MM:SS, gdzie HH to godzina, MM minuty oraz SS to sekundy.

Typy związków zachodzących pomiędzy tabelami (encjami):

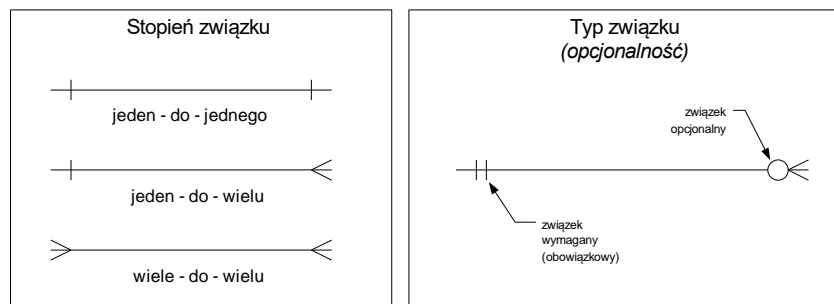
- 1:1 ("jeden do jeden") – encji odpowiada dokładnie jedna encja,

- 1:N ("jeden do wielu") – encji odpowiada jedna lub więcej encji,
- M:N ("wiele do wielu") – jednej lub więcej encjom odpowiada jedna lub więcej encji.

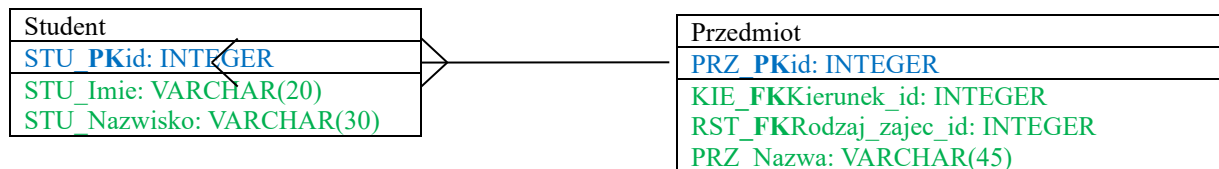
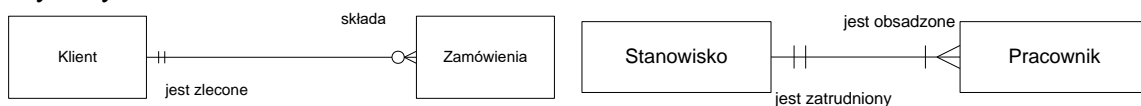
Opcjonalność dotyczy zaangażowania encji w związek. Z uwagi na tę cechę wyróżnia się dwa typy związków:

wymagane (obowiązkowe) – zachodzi wówczas, jeśli wszystkie wystąpienia encji muszą uczestniczyć w związku;

opcjonalne - zachodzi wówczas, jeśli istnieje, co najmniej jedno wystąpienie encji, które nie uczestniczy w związku.



Przykłady:



W praktyce stosowanie tejże relacji jest wielce niewskazane ze względu na redundancję danych, zatem rozwiązaniem tego problemu będzie utworzenie tzw. tabeli łącznikowej, scalającej obie tabele relacją *jeden-do-wielu*.

