

# Bazy danych 2022 - Laboratoria:

## Model konceptualny

Bartosz Brzostowski

19 kwietnia 2022

### 1 Krótkie przypomnienie podstawowych informacji o relacyjnych bazach danych

Dane w bazie są przechowywane w formie relacji, które z perspektywy użytkownika mają postać tabel. Każda relacja składa się z krotek (czyli rekordów albo wierszy) oraz atrybutów (pól lub kolumn).

**Tabele** są głównymi strukturami w bazie danych. Każda tabela zawsze odzwierciedla jedno konkretne zagadnienie. Każda tabela zawiera przynajmniej jedną kolumnę — zwaną kluczem głównym — która pozwala w jednoznaczny sposób zidentyfikować każdy wiersz.

**Kolumna** odzwierciedla pewną cechę zagadnienia reprezentowanego przez tabelę. Kolumny są strukturami służącymi do przechowywania danych. Dane zawarte w kolumnach można odczytywać oraz prezentować w postaci konkretnych informacji. Jakość informacji otrzymanych na podstawie danych zależy od integralności samych danych w kolumnach. Każda kolumna w poprawnie zaprojektowanej bazie danych zawiera jedną i tylko jedną wartość, a nazwa kolumny odzwierciedla rodzaj tej wartości. Dzięki temu wprowadzanie danych do kolumn jest intuicyjne. Przykładowo dokładnie wiadomo jakiego rodzaju dane powinny trafić do kolumn o nazwach takich jak *Miasto*, *KodPocztowy* czy *Imie* (Uwaga - w celu uniknięcia ewentualnych kłopotów z obsługą polskich znaków często przyjmuje się, że nazwy kolumn w tabelach nie powinny ich zawierać w ogóle).

**Wiersz** reprezentuje unikatowe wystąpienie (instancję) zagadnienia, które ilustruje tabela. Wiersz składa się z całego zestawu kolumn tabeli, niezależnie od tego, czy poszczególne kolumny zawierają jakieś dane. Ze względu na sposób zdefiniowania tabeli każdy rekord w bazie danych jest identyfikowany przy użyciu unikatowej wartości zapisanej w kolumnie (lub kolumnach) klucza głównego danego wiersza.

**Klucze** to specjalne pola, które odgrywają w tabeli szczególną rolę. O tym, jaką funkcję pełni dany klucz w tabeli, decyduje jego rodzaj. Klucz główny to kolumna lub kilka kolumn, które w jednoznaczny sposób identyfikują każdy wiersz w tabeli. (Jeśli klucz główny składa się z dwóch lub większej liczby kolumn, to mówimy o złożonym kluczu głównym). Priorytetowa rola klucza głównego bierze się z dwóch powodów: jego wartość identyfikuje konkretny wiersz w całej bazie danych, zaś jego kolumna identyfikuje daną tabelę w całej bazie danych.

Klucze główne wymuszają integralność bazy na poziomie tabel i ułatwiają definiowanie relacji między tabelami. Każda tabela w bazie danych powinna mieć klucz główny.

Jeśli między dwiema tabelami zachodzi jakaś zależność, to na ogół definiuje się ją tak,

że bierze się kopię klucza głównego z pierwszej tabeli i umieszcza ją w drugiej tabeli, w której pełni on wówczas funkcję klucza obcego. (Określenie klucza obcy wywodzi się z faktu, że druga tabela ma już swój własny klucz główny, a drugi klucz, zapożyczony z pierwszej tabeli, dla tej drugiej jest „obcy”). Klucze obce są istotne nie tylko ze wspomnianego oczywistego powodu, że ułatwiają ustanawianie zależności między parami tabel, ale także pomagają w zachowaniu integralności struktury danych na poziomie tych zależności. To oznacza, że wiersze w obu tabelach zawsze będą poprawnie powiązane: wartości klucza obcego muszą odpowiadać wartościom klucza głównego, do którego się odwołują. Klucze obce pozwalają zarazem uniknąć niepożądanych „osieroconych wierszy”.

**Widok** to wirtualna tabela składająca się z kolumn pochodzących z jednej lub kilku tabel znajdujących się w bazie danych. Tabele wchodzące w skład widoku nazywa się tabelami bazowymi. Model relacyjny traktuje widoki jako twory wirtualne, ponieważ zawarte w nich dane są czerpane z tabel bazowych i nie są przechowywane nigdzie indziej. Tak naprawdę jedyną informacją o widoku zapisaną w bazie danych to jego struktura. Widoki umożliwiają przeglądanie informacji znajdujących się w bazie z różnych perspektyw, co zdecydowanie uelastycznia możliwości pracy z danymi. Widoki można tworzyć na wiele sposobów — te najbardziej przydatne bazują na kilku powiązanych tabelach.

## 1.1 Zależności

Jeśli wiersze z danej tabeli można w jakiś sposób powiązać z wierszami w innej tabeli, to znaczy, że między tymi tabelami występuje pewna zależność (potocznie zwana relacją albo związkiem). Sposób zdefiniowania tej zależności jest uzależniony od jej rodzaju. Między dwiema tabelami można wyróżnić trzy rodzaje zależności: *jeden do jednego*, *jeden do wielu* oraz *wiele do wielu*.

- **Jeden do jednego**

Związek jeden do jednego między dwiema tabelami zachodzi wówczas, gdy jeden wiersz z pierwszej tabeli jest powiązany z tylko jednym wierszem z drugiej tabeli, zaś jeden wiersz z drugiej tabeli jest powiązany z tylko jednym wierszem z pierwszej tabeli. Ten rodzaj relacji może być postrzegany jako podzielenie tabeli na dwie (bo relacja jest jeden do jeden). Utworzenie zależności polega na wzięciu klucza głównego z tabeli jednej tabeli i wstawieniu go do drugiej tabeli, gdzie staje się on kluczem obcym. Często klucz obcy pełni zarazem funkcję klucza głównego tabeli drugorzędnej. Wybór tabeli, która w zależności jeden do jednego będzie pełniła funkcję tabeli głównej, jest uzależniony od tego, czy wiersze zawarte w jednej tabeli mogą istnieć bez pasujących do nich wierszy w drugiej tabeli. W zależności jeden do jednego nie da się dodać wiersza do tabeli podrzędnej, jeśli w tabeli głównej nie istnieje już pasujący do niego wiersz. Związki jeden do jednego nie są powszechne

- **Jeden do wielu**

Gdy w dwóch tabelach można wyróżnić zależność jeden do wielu, to znaczy, że jeden wiersz z pierwszej tabeli może być powiązany z wieloma wierszami z drugiej tabeli, ale jeden wiersz z drugiej tabeli może być powiązany z tylko jednym wierszem z pierwszej tabeli. Tę zależność definiuje się poprzez wzięcie klucza głównego z tabeli po stronie „jeden” i wstawienie go do tabeli po stronie „wiele”, gdzie staje się on kluczem obcym.

- **Wiele do wielu**

Między dwiema tabelami można wyróżnić zależność wiele do wielu, jeśli jeden wiersz

z pierwszej tabeli może być powiązany z wieloma wierszami z drugiej tabeli, zaś jeden wiersz z drugiej tabeli może być powiązany z wieloma wierszami z pierwszej tabeli. W celu prawidłowego zdefiniowania takiej relacji trzeba utworzyć tabelę wiążącą (pośrednią). Ta tabela stanowi wygodny sposób na powiązanie wierszy z jednej tabeli z tymi z drugiej tabeli i zarazem ułatwia dodawanie, usuwanie lub modyfikowanie powiązanych danych. W celu zdefiniowania tabeli wiążącej należy wziąć kopie kluczy głównych z obydwu tabel i użyć ich do utworzenia struktury nowej tabeli. Kolumny w tej tabeli będą pełniły dwojaką funkcję: łącznie będą stanowiły złożony klucz główny tabeli wiążącej, zaś osobno będą odgrywały rolę kluczy obcych.

## 2 Prawidłowa struktura bazy danych

W bazie danych, której struktura nie jest w pełni poprawna, będą pojawiały się problemy z odczytywaniem z bazy nawet pozornie prostych informacji; trudno będzie pracować z danymi, a konieczność dodania albo usunięcia jakiejś kolumny z tabeli będzie uciążliwa.

### Kolumny

Przy wybieraniu nazw dla kolumn należy zwrócić uwagę na następujące aspekty (nieuwzględnienie ich może powodować problemy przy pracy z bazą danych):

- Czy na podstawie nazwy można łatwo określić, jakie cechy dotyczy. Nazwy niejednoznaczne i niejasne zwiastują kłopoty i sugerują, że cel utworzenia kolumny nie został dobrze przemyślany.
- Czy nazwa jest rzeczowa i zrozumiała dla innych – jeśli z bazą danych będą pracowali różni użytkownicy to nazwa ma być zrozumiała dla wszystkich, którzy będą używali danej kolumny. Użycie słowa, które będzie miało różne znaczenie dla różnych grup ludzi będzie powodować kłopoty.
- Czy nazwa kolumny jest jednoznaczna? Przykładowo nazwa kolumny NumerTelefonu może być myląca, gdyż nie wiadomo czy chodzi nr telefonu do pracy, nr domowy czy komórkowy. Należy precyzyjnie dobierać nazwy kolumn – jeśli w bazie mają być przechowywane różne rodzaje numerów telefonu to trzeba utworzyć odpowiednie pola np. TelefonDomowy, TelefonFirmowy i TelefonKomorkowy.
- Dobrze jest zadbać o to by ta sama nazwa kolumny nie powtarzała się w różnych tabelach. Można to zrobić przez dodanie prefiksu lub sufiksu. Przypuśćmy, że w tabelach o nazwach Klienci i Dostawcy przechowywany ma być adres. Nadanie nazw AdresKlienci czy DostawcyAdres pozwoli jednoznacznie rozróżniać te kolumny z różnych tabel. Wyjątkiem od tej reguły może być użycie kolumny w celu zdefiniowania zależności między dwiema tabelami. Powszechność zastosowania prefiksów w nazwach pól w tabeli to kwestia przyjętej konwencji – ważne jest o konsekwentne przestrzeganie jej w całej strukturze bazy.
- Należy unikać używania akronimów lub skrótów na nazwy kolumny. Zbyt skrócone nazwy mogą być trudne do rozszyfrowania i łatwo wprowadzają w błąd.
- Nazwa kolumny powinna odnosić się do jednej cechy. Nazwa, która pośrednio lub bezpośrednio odwołuje się do więcej niż jednej cechy zwykle zawiera spójniki „i” albo „lub”. Podobnie rzecz się ma z kolumnami zawierającymi ukośnik (\), myślnik

(–) albo ampersand (&). Jeśli w tabeli znajdują się kolumny o nazwach takich jak np. Telefon\Fax, to lepiej podzielić je na mniejsze, bardziej precyzyjnie zdefiniowane kolumny.

- Według standardu SQL prawidłowy identyfikator ma nazwę zaczynającą się od litery i zawierającą tylko litery, cyfry oraz znak podkreślenia. Spacje nie są dozwolone. Standard ten przewiduje także użycie nazw w postaci identyfikatorów z ogranicznikami – takie nazwy są ujęte w podwójny cudzysłów, muszą zaczynać się od litery i mogą zawierać litery, cyfry, znaki podkreślenia, spacje oraz znaki specjalne z pewnego ściśle określonego zbioru. W odniesieniu do nazw kolumn zalecane jest jednak używanie tylko pierwszej konwencji (część implementacji SQL obsługuje wyłącznie takie identyfikatory).
- Wskazane jest stosowanie rzeczowników w liczbie pojedynczej na nazwy kolumn. Z kolei nazwa tabeli powinna na ogół mieć formę w liczbie mnogiej, ponieważ stanowi ona zbiór wielu podobnych obiektów albo zdarzeń.
- Należy upewnić się, że kolumna odzwierciedla konkretną cechę zagadnienia reprezentowanego przez tabelę. Chodzi o to, by się upewnić, że dana kolumna rzeczywiście powinna należeć do tej tabeli. Jeśli nie ma ona związku z tematem tabeli, usuń ją albo ewentualnie przenieś do innej tabeli. Wyjątek od tej reguły dotyczy sytuacji, gdy dana kolumna jest używana do zdefiniowania zależności między daną tabelą a innymi tabelami w bazie.
- Kolumna powinna zawierać tylko jedną wartość. Kolumna, która może potencjalnie zawierać kilka wystąpień wartości tego samego typu, nosi nazwę kolumny wielowartościowej (np. kolumna zawierająca kilka numerów telefonów). Na tej samej zasadzie kolumna, w której potencjalnie można zapisać dwie lub większą liczbę wartości różnych typów, nosi nazwę wieloczęściowej (kolumna, która zawiera kod pocztowy oraz miasto jest przykładem kolumny wieloczęściowej). Kolumny wielowartościowe i wieloczęściowe mogą przysparzać niemałych problemów, zwłaszcza podczas prób edytowania, usuwania albo sortowania danych.
- Kolumna nie powinna zawierać wyniku obliczeń albo złączenia. Problem jest związany z wartością takiej kolumny – nie zawiera ona wzoru obliczeń. Gdy wartość dowolnego składnika obliczeń ulegnie zmianie, rezultat przechowywany w kolumnie nie zostanie zaktualizowany.
- Należy dbać o spójność danych – dana kolumna powinna pojawiać się w całej bazie tylko jeden raz.

### **Eliminowanie kolumn wieloczęściowych**

Podzielić zawartość kolumny na bardziej precyzyjne części (odrębne kolumny). Czasami rozpoznanie kolumny wieloczęściowej może być trudne. Przykładowo w kolumnie Sprzęt przechowującej dane o aparaturze z laboratorium mogą być przechowywane pozycje Wzmacniacz MZ100 czy Zasilacz ZTX. Jest to w istocie kolumna wieloczęściowa – przechowywane są dane o rodzaju urządzenia (wzmacniacz, zasilacz) oraz informacje o jego modelu czy typie.

### **Eliminowanie pól wielowartościowych**

Identyfikacja kolumn wielowartościowych jest prosta – na ogół dane przechowywane w tego

rodzaju kolumnach zawierają przecinki, średniki albo inne znaki często używane w charakterze separatorów. Eliminowanie kolumn wielowartościowych jest nieco bardziej skomplikowane. Należy przeanalizować zależność między tą kolumną a tabelą, w której została ona umieszczona. Wartości w kolumnie wielowartościowej mogą pozostawać w związku wiele do wielu z każdym z wierszy w macierzystej tabeli czyli dowolna wartość z kolumny wielowartościowej może być powiązana z wieloma wierszami z tabeli macierzystej, a jeden wiersz z tabeli macierzystej może być powiązany z dowolną liczbą wartości w kolumnie wielowartościowej. Tego rodzaju zależność wiele do wielu należy potraktować tak samo jak w przypadku innych związków tego typu tj. utworzyć dla niej tabelę łączącą. Taką tabelę łączącą można utworzyć przy użyciu kolumny wielowartościowej oraz kopii kolumny klucza głównego z tabeli macierzystej — te dwa elementy staną się podstawą nowej tabeli.

### Tabele

Tabele są podstawą wszystkich zapytań SQL. Źle zaprojektowane tabele mogą się przyczyniać do problemów z integralnością danych czy też przysparzać trudności przy tworzeniu zapytań SQL obejmujących wiele tabel. To sprawia, że naprawdę warto zadbać o efektywną strukturę tabel, by pozyskiwanie potrzebnych danych było jak najłatwiejsze.

Podobnie jak dla kolumn dobór nazw tabel warto przemyśleć. Tabela powinna reprezentować jedno zagadnienie. Jeśli zawiera ona dane dotyczące różnych zagadnień, to należy ją podzielić na mniejsze tabele. Nazwa tabeli musi jasno odzwierciedlać jej temat. Enigmatyczna albo niejednoznaczna nazwa tabeli świadczy o tym, że jej struktura nie została dobrze przemyślana. Przy wybieraniu nazw dla tabel należy zwrócić uwagę na następujące aspekty:

- Czy nazwa jest unikatowa i wystarczająco opisowa, by była zrozumiała dla innych użytkowników.
- Czy nazwa precyzyjnie, jasno i jednoznacznie odzwierciedla temat tabeli? Jeśli nazwa tabeli jest niejasna albo wieloznaczna, to można z dużym prawdopodobieństwem założyć, że tabela ta reprezentuje kilka zagadnień.
- Czy nazwa zawiera słowa opisujące fizyczne właściwości danych? Należy unikać używania słów takich jak Plik, Rekord albo Tabela w nazwach tabel, ponieważ wprowadzają one zamieszanie.
- Czy nazwa tabeli jest akronimem albo skrótem? Skróty rzadko dobrze oddają tematykę tabeli, a akronimy są na ogół trudne do rozszyfrowania.
- Czy użyta nazwa pośrednio lub bezpośrednio odwołuje się do więcej niż jednego tematu? Jest to jeden z częstszych błędów przy nazywaniu tabel, który zarazem dość łatwo wychwycić. Tego rodzaju nazwy zwykle zawierają spójniki „i” albo „lub” oraz znaki takie jak ukośnik (\), myślnik (–) albo ampersand (&).
- Wskazane jest nazywanie tabel w liczbie mnogiej jest zalecane, ponieważ przechowują one zbiór wystąpień zagadnienia, któremu są poświęcone.
- Tabela powinna reprezentować tylko jedno zagadnienie – zmniejsza to ryzyko potencjalnych problemów z integralnością danych.
- Każda tabela powinna mieć klucz główny. Trzeba to zrobić z dwóch powodów. Po pierwsze, klucz główny jednoznacznie identyfikuje każdy wiersz w tabeli, po drugie

zaś, służy on do definiowania związków między tabelami. Jeśli nie będzie zdefiniowany klucz główny we wszystkich tabelach, to wystąpią problemy z integralnością danych, a także z realizacją niektórych zapytań SQL, obejmujących wiele tabel.

**Klucze** Klucz główny jest jednym z najważniejszych kluczy w tabeli, ponieważ pozwala on na zidentyfikowanie każdego wiersza w tabeli, jak również identyfikuje samą tabelę w bazie danych. Klucz tego rodzaju pozwala też na ustanowienie związku między dwiema tabelami. Każda tabela w bazie powinna zawierać taki klucz!

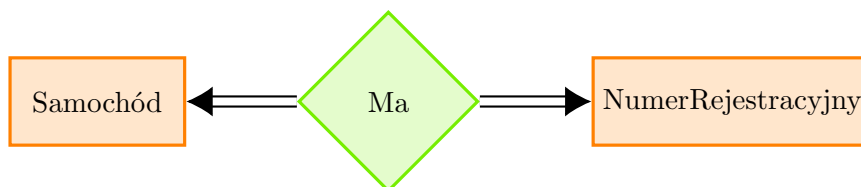
Klucz główny jest kolumną albo grupą kolumn jednoznacznie identyfikującą każdy wiersz w tabeli. Klucz główny składający się z jednej kolumny jest nazywany kluczem głównym prostym (albo zwyczajnie kluczem głównym). Klucz główny składający się z dwóch lub większej liczby kolumn nazywa się kluczem głównym złożonym. Zawsze gdy to tylko możliwe, należy definiować klucze główne proste, ponieważ tworzenie związków za ich pomocą jest bardziej efektywne i łatwiejsze. Klucze złożonych należy używać tam, gdzie to konieczne, na przykład przy tworzeniu i definiowaniu tabeli łączącej.

Do utworzenia klucza głównego może posłużyć dowolna kolumna lub kombinacja kolumn spełniająca podane niżej kryteria:

- Czy kolumny klucza w sposób jednoznaczny identyfikują dowolny wiersz w tabeli? Każdy wiersz w tabeli reprezentuje pewne wystąpienie tematu tej tabeli. Dobry klucz główny gwarantuje możliwość zidentyfikowania lub odwołania się do każdego wiersza z tej tabeli z poziomu innych tabel w bazie danych. Pomaga on też w uniknięciu duplikowania wierszy w tabeli.
- Czy wybrana kolumna lub kombinacja kolumn zawiera unikatowe wartości? Dopóki wartości klucza głównego są unikatowe, istnieje możliwość sprawdzenia, że w tabeli nie ma powtarzających się wierszy.
- Czy wybrane kolumny będą mogły zawierać nieznane wartości? To bardzo ważne pytanie, ponieważ klucz główny nie może zawierać nieznanych wartości. Kolumnę, w której mogą się kiedyś znaleźć niesprecyzowane wartości, należy od razu zdyskwalifikować.
- Czy wartość wybranych kolumn może być opcjonalna? Jeśli tak, to nie należy używać takich kolumn w charakterze klucza głównego. Opcjonalna wartość kolumny oznacza, że na pewnym etapie mogą się w niej pojawić nieznane wartości.
- Czy wartości w wybranych kolumnach mogą kiedyś ulec zmianie? Wartości w kolumnach klucza głównego powinny pozostać stałe. Nigdy nie należy zmieniać wartości w kolumnie klucza głównego; chyba że ma się wyjątkowy powód. Jeśli wartość kolumny może podlegać arbitralnym zmianom, to taka kolumna raczej nie będzie spełniała warunków podanych powyżej.
- W sytuacji, gdy w tabeli nie ma kolumny, której można by użyć jako klucza głównego, gdyż wartość każdej z zawartych w niej kolumn może ulegać arbitralnym zmianom, należy utworzyć sztuczny klucz główny. Jest to arbitralna kolumna, którą należy zdefiniować i dodać do tabeli tylko po to, by można było używać jej jako klucza głównego.

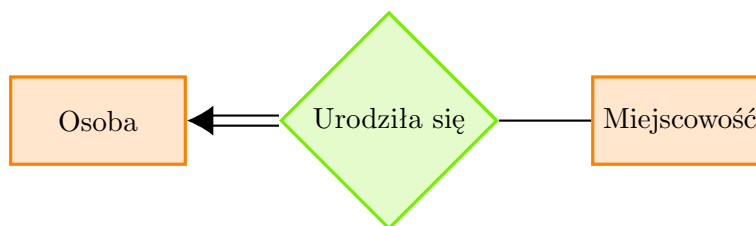
### 3 Definiowanie zależności

**Związek jeden do jednego** definiuje się poprzez wzięcie klucza głównego z tabeli głównej i wstawienie go do tabeli podrzędnej, w której staje się on kluczem obcym. Jest to szczególny rodzaj zależności, ponieważ w wielu przypadkach klucz obcy będzie zarazem pełnił funkcję klucza głównego tabeli podrzędnej.



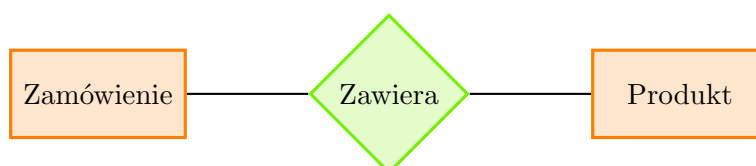
Każdy (używany) samochód ma numer rejestracyjny. Każdy (używany) numer rejestracyjny jest przypisany do konkretnego samochodu.

**Związek jeden do wielu** definiuje się poprzez wzięcie klucza głównego z tabeli po stronie „jeden” i wstawienie go do tabeli po stronie „wiele”, gdzie staje się on kluczem obcym.



Każda osoba urodziła się w konkretnej miejscowości. W danej miejscowości mogło urodzić się wiele osób.

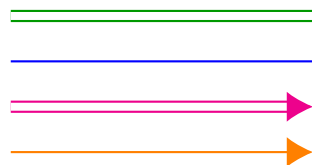
**Związek wiele do wielu** definiuje się przez utworzenie tabeli łączącej. Tabelę taką tworzy się na podstawie kopii kluczy głównych należących do każdej z powiązanych tabel. Kolumny te zwykle pełnią dwie odrębne funkcje: łącznie stanowią złożony klucz główny tabeli łączącej, zaś osobno są kluczami obcymi.



Zamówienie może zawierać wiele produktów. Produkt może występować w wielu zamówieniach.

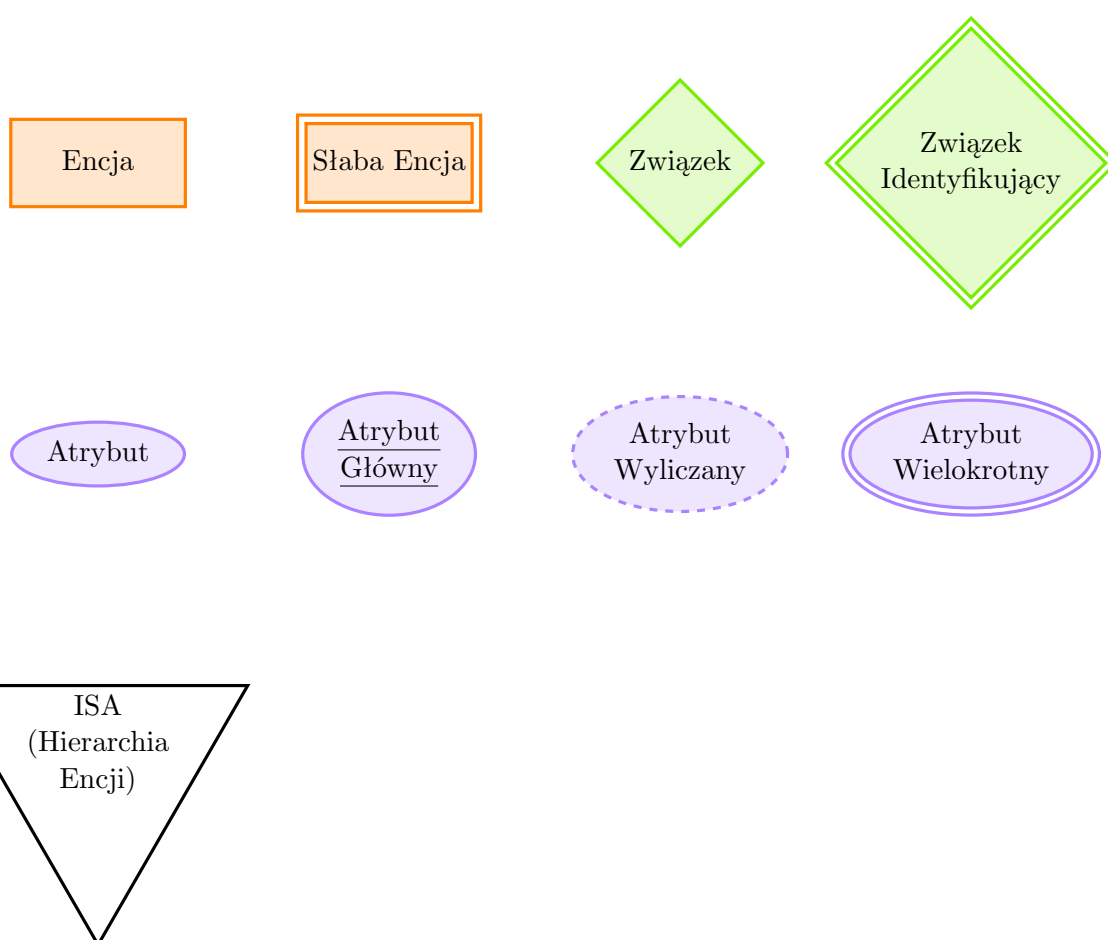
### 4 Notacja Chena

Można napotkać różne warianty notacji Chena (położenie strzałek, typy linii). Proszę stosować notację przedstawioną na wykładzie nr 5. Jeśli chodzi o typy linii to proszę używać następujących:



Linia pojedyncza związek opcjonalny, a linia podwójna związek obowiązkowy (po stronie encji, dla której związek jest obowiązkowy). W celu uniknięcia nieporozumień **proszę NIE używać linii pogrubionej dla związku obowiązkowego**. Strzałka oznacza związek jeden do wielu, przy czym ma być po stronie „jeden”, czyli naprzeciw encji, która najwyżej raz wchodzi w związek.

**Symbole używane w notacji Chena:**




---

W razie pytań czy wątpliwości proszę kontaktować się z prowadzącym lub wykładowcą. Więcej informacji na temat modelowania znajduje się w wykładzie nr 5. A jeszcze więcej np. [WWW.GOOGLE.COM](http://WWW.GOOGLE.COM).