



# Bazy danych 2022

## Wykład 1

### Podstawy modelu relacyjnego

Bartosz Brzostowski

Wydział Fizyki i Astronomii UWr  
semestr letni r. akad. 2021/22

3 marca 2022



## ► Literatura

- J. D. Ullman, J. Widom, *Podstawowy kurs systemów baz danych*, WNT, 1999
- H. Garcia-Molina, J. D. Ullman, J. Widom, *Implementacja systemów baz danych*, WNT, 2003
- T. Connolly, C. Begg, *Database Systems*, Addison Wesley, 2002; po polsku: ReadMe, 2004
- C. J. Date, *Wprowadzenie do systemów baz danych*, WNT, 2000
- T. Converse, J. Park, C. Morgan, *PHP5 i MySQL. Biblia*, Helion, 2005
- [http://wazniak.mimuw.edu.pl/index.php?title=Bazy\\_danych](http://wazniak.mimuw.edu.pl/index.php?title=Bazy_danych) ← UWAGA! Przykłady są w ORACLE SQL, a nie w MySQL
- [http://smurf.mimuw.edu.pl/node/3?q=bazy\\_danych](http://smurf.mimuw.edu.pl/node/3?q=bazy_danych)
- *MySQL 5.7 Reference Manual*  
<https://dev.mysql.com/doc/refman/5.7/en/>
- Zasady zaliczania → [1.Wstęp.pdf](#) w materiałach do zajęć dla zespołu utworzonego w MS Teams dla **wykładu**.

Dotychczas wydanych zostało m n ó s t w o książek dotyczących MySQL i PHP. Nie sposób przejrzeć i ocenić wszystkie. Także oceny na stronach wydawnictw takich jak *Helion* bywają mocno rozbieżne. Przykładowe pozycje:

- ▶ A. Beaulieu, *Wprowadzenie do SQL. Jak generować, pobierać i obsługiwać dane*, Helion, 2021
- ▶ J. L. Viescas, *Zapytania w SQL. Przyjazny przewodnik*, Helion, 2020
- ▶ L. Welling, L. Thomson, *PHP i MySQL. Tworzenie stron WWW. Vademecum profesjonalisty*, Helion, 2017
- ▶ M. Lis, *PHP i MySQL. Dla każdego*, Helion, 2017
- ▶ M. Lis, *SQL. Ćwiczenia praktyczne*, Helion, 2014

Także w internecie można znaleźć ogromną ilość kursów, poradników etc., należy jednak korzystać z nich z rozwagą, weryfikując kiedy powstały i której implementacji SQL dotyczą.



- Lata 60. Pierwsze rozwiązania (proto?-)bazodanowe
- Lata 70. Model relacyjny — uporządkowanie danych w tabelach, formalny, teoretyczny opis  
Wczesne implementacje, np. Ingres, System R
- Lata 90. RDBMS trafia pod strzechy — powstają MySQL, PostgreSQL    [RDBMS → Relational Database Management System](#)
- XXI w. Modne stają się nowe, wyspecjalizowane rozwiązania, zorientowane np. na przetwarzanie w chmurze, inne modele danych  
Zmierzch RDBMS? Raczej nie, dalej używają ich również giganci (Google, Facebook)

# Relacje? Jakie relacje?

- ▶ Teoria zbiorów: relacja to dowolny podzbiór jakiegoś iloczynu kartezjańskiego zbiorów

- ▶ Rozważmy iloczyn ( $\Sigma$  to alfabet łaciński)

zbiór "numerów indeksów"

$$\{n \in \mathbb{N} : 100000 \leq n < 1000000\} \times \Sigma^* \times \Sigma^*$$

zbiór wszystkich słów  $\Sigma^*$   
("dwie kopie")

- ▶ Jego podzbiorem jest na przykład

indeks	imię	nazwisko
285456	Jan	Kowalski
250964	Karolina	Nowak
294606	Maciej	Bąk
...		

- ▶ Czyli relacje to po prostu tabele!



- ▶ (Co najmniej) dwa zestawy pojęć
- ▶ Bardziej „teoretycznie”: *relacje* mają *atrybuty*, które mają określone *dziedziny*, oraz zawierają *krotki*
- ▶ Bardziej „praktycznie”: *tabele* mają *kolumny* określonego *typu (danych)*, oraz zawierają *wiersze*
- ▶ Schemat tabeli: jej nazwa oraz nazwy i typy kolumn
- ▶ Stan tabeli: wiersze, które zawiera
- ▶ Zmiana stanu nie zmienia schematu
- ▶ Cała baza, jako zestaw tabel, też ma schemat i stan
- ▶ Skróty
  - ang. Database    pol. Baza Danych
  - DBMS → Database Management System
  - ▶ baza danych: DB, BD    SZBD → System Zarządzania Bazą Danych
  - ▶ system zarządzania bazą danych: DBMS, SZBD
  - ▶ SZ relacyjną BD: RDBMS, SZRBD

► Typowa architektura aplikacji bazodanowej:

1. Baza danych

2. DBMS

3. Aplikacja

4. HTTP(S)

5. Przeglądarka internetowa użytkownika

(możliwe warianty, np. dedykowane oprogramowanie sieciowe zamiast 3–5)

- Nawet jeśli trzeba gruntownie zmienić aplikację, to dobrze zaprojektowana baza to ułatwi
- Zmiana DBMS: eksport / import bazy danych... i walka z niekompatybilnością

Tych elem.  
(raczej) nie  
będziemy  
reimple-  
mentować

Nad tymi elementami mamy pełną kontrolę  
i musimy dobrze je zaprojektować

- ▶ Czy „goły” zbiór tabel jest już bazą danych?
- ▶ Technicznie tak. . .
- ▶ Modelowana rzeczywistość ma swoje reguły
  - ▶ Nie ma dwóch samochodów o tym samym numerze rejestracyjnym
  - ▶ Post na forum zawsze ma swojego autora
- ▶ Bazy danych też mają reguły — więzy
- ▶ Jak zdefiniujemy je w bazie, nie będzie trzeba już się o nie martwić na poziomie aplikacji — zadba o nie DBMS
- ▶ Rodzaje: kolumnowe, tabelowe, między tabelami
- ▶ Również stanowią element schematu bazy — a stan bazy musi je spełniać





- ▶ Kolumna `nr_lokalu` w tabeli adres
- ▶ A co, jeśli ktoś mieszka w domku?
- ▶ „Specjalna” wartość? Liczba ujemna, pusty napis?
- ▶ Lepiej: *brak* wartości
- ▶ NULL nie jest wartością, tylko stanem

```
mysql> SELECT NULL = NULL;  
NULL
```

```
mysql> SELECT NULL IS NULL;  
1
```

```
mysql> SELECT 0 IS NULL;  
0
```

To jest operator

```
mysql> SELECT 0 IS 0;  
ERROR 1064 (42000): You have an error in your SQL  
syntax; check the manual that corresponds to [...]
```

A to jest błąd składniowy

- ▶ *NULL-safe comparison*: `<=>`

W obliczeniach zmiennoprzecinkowych (i nie tylko) analogicznym pojęciem jest NaN ("not a number"), które z kolei jest analogiczne do symbolu nieoznaczonego w analizie matematycznej.

Np. gdy  $n \rightarrow \infty$  wtedy

$\frac{1}{n} = 1 \rightarrow 1$ ,  $\frac{2}{n} = 2 \rightarrow 2$ ,  
ale oba ułamki "są równe"  $\frac{\infty}{\infty}$ .

Nie można więc stwierdzić, że

$\frac{\infty}{\infty} = \frac{\infty}{\infty} \dots$



► Logika trójwartościowa: prawda, fałsz, NULL

```
mysql> SELECT NOT NULL;  
NULL
```

```
mysql> SELECT TRUE OR NULL;  
1
```

```
mysql> SELECT FALSE OR NULL;  
NULL
```

```
mysql> SELECT TRUE AND NULL;  
NULL
```

```
mysql> SELECT FALSE AND NULL;  
0
```

Przyjmujemy, że TRUE = 1, FALSE = 0, a NULL to ... nie wiadomo co, ale więcej niż 0 i mniej niż 1. Czyli przyjmujemy, że NULL jest równoważny x przy czym zachodzi nierówność  $0 < x < 1$ .

Wtedy AND = MIN, natomiast OR = MAX.

MIN(0,x) = 0    FALSE AND NULL = FALSE

MIN(1,x) = x    TRUE AND NULL = NULL

MAX(0,x) = x    FALSE OR NULL = NULL

MAX(1,x) = 1    TRUE OR NULL = TRUE

► NULL propagation

```
mysql> SELECT 1 + NULL;  
NULL
```

```
mysql> SELECT CONCAT("sklejone", NULL, "napisy");  
NULL
```

- ▶ Zbiór kolumn (być może jednoelementowy), których stan jednoznacznie identyfikuje wiersz (wraz ze stanem pozostałych kolumn)
  - ▶ Numer rejestracyjny pojazdu identyfikuje pojazd
  - ▶ Dzień tygodnia, godzina rozpoczęcia i sala identyfikują pozycję w planie lekcji
- ▶ Trywialny klucz — wszystkie kolumny tabeli
- ▶ Klucz kandydujący — minimalny taki zbiór
- ▶ Jeden z nich można wybrać jako klucz główny
- ▶ Klucze proste i złożone (patrz wyżej)
- ▶ Klucze naturalne i sztuczne — „wewnętrzne” ID w bazie



- ▶ Kolumnowe — NOT NULL, UNIQUE, PRIMARY KEY  
Uwaga: kolumna UNIQUE może zawierać więcej niż jeden NULL, bo... zdanie „ $\text{NULL} \neq \text{NULL}$ ” nie jest nieprawdziwe (tylko NULLowe)!
- ▶ Tabelowe — to samo (poza NOT NULL) gdy dotyczy więcej niż jednej kolumny
- ▶ Pomiędzy tabelami — klucze obce: wartości w kolumnie wskazują na (tj. muszą występować w) kolumnę-klucz innej tabeli
- ▶ Związki, nie relacje (bo relacje to tabele)



- ▶ Co będzie?
  - ▶ SQL — te 20%, których używa się w 80% sytuacji
  - ▶ modelowanie i normalizacja baz danych
  - ▶ HTML i PHP — tyle, żeby napisać prostą aplikację
- ▶ Czego (prawie) nie będzie?
  - ▶ Wszystkiego, co interesujące i/lub życiowe. . .
  - ▶ Teoria baz danych
  - ▶ Elementy proceduralne w SQL, wyzwalacze
  - ▶ Przetwarzanie transakcyjne, współbieżność
  - ▶ Zarządzanie prawami dostępu (DCL)
  - ▶ Bezpieczeństwo aplikacji
  - ▶ Rozproszone bazy danych, partycjonowanie
  - ▶ Specjalistyczne, nierelacyjne bazy danych (NoSQL)