

# BAZY DANYCH

teoria

- Baza danych (BD) — zbiór powiązanych ze sobą danych
- System zarządzania bazą danych (SZBD) — zestaw narzędzi programowych do zarządzania danymi

*Database Management System,  
DBMS*

odpowiada za:

- *obsługę języka bazy danych*
- *przechowywanie i pobieranie danych*
- *optymalizację BD*

- Plik – Serwer: MS Access

Pliki danych są centralnie przechowywane na serwerze plików, natomiast SZBD jest zainstalowany na każdym komputerze klienckim.

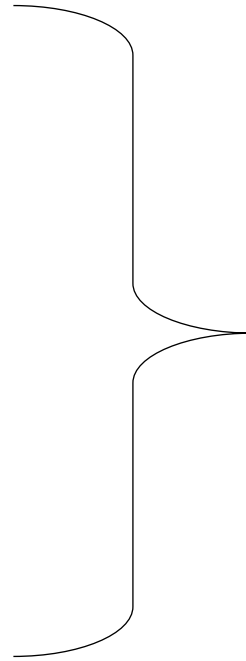
- Klient – Serwer: MySQL, PostgreSQL...

Zarówno SZBD, jak i sama baza danych znajdują się na serwerze i są dostępne zdalnie z komputerów klienckich

- Wbudowane: SQLite

lżejsza wersja SZBD często używana jako część aplikacji mobilnej.

- MySQL
- PostgreSQL
- Oracle
- Ms SQL
- Maria DB (~ MySQL)



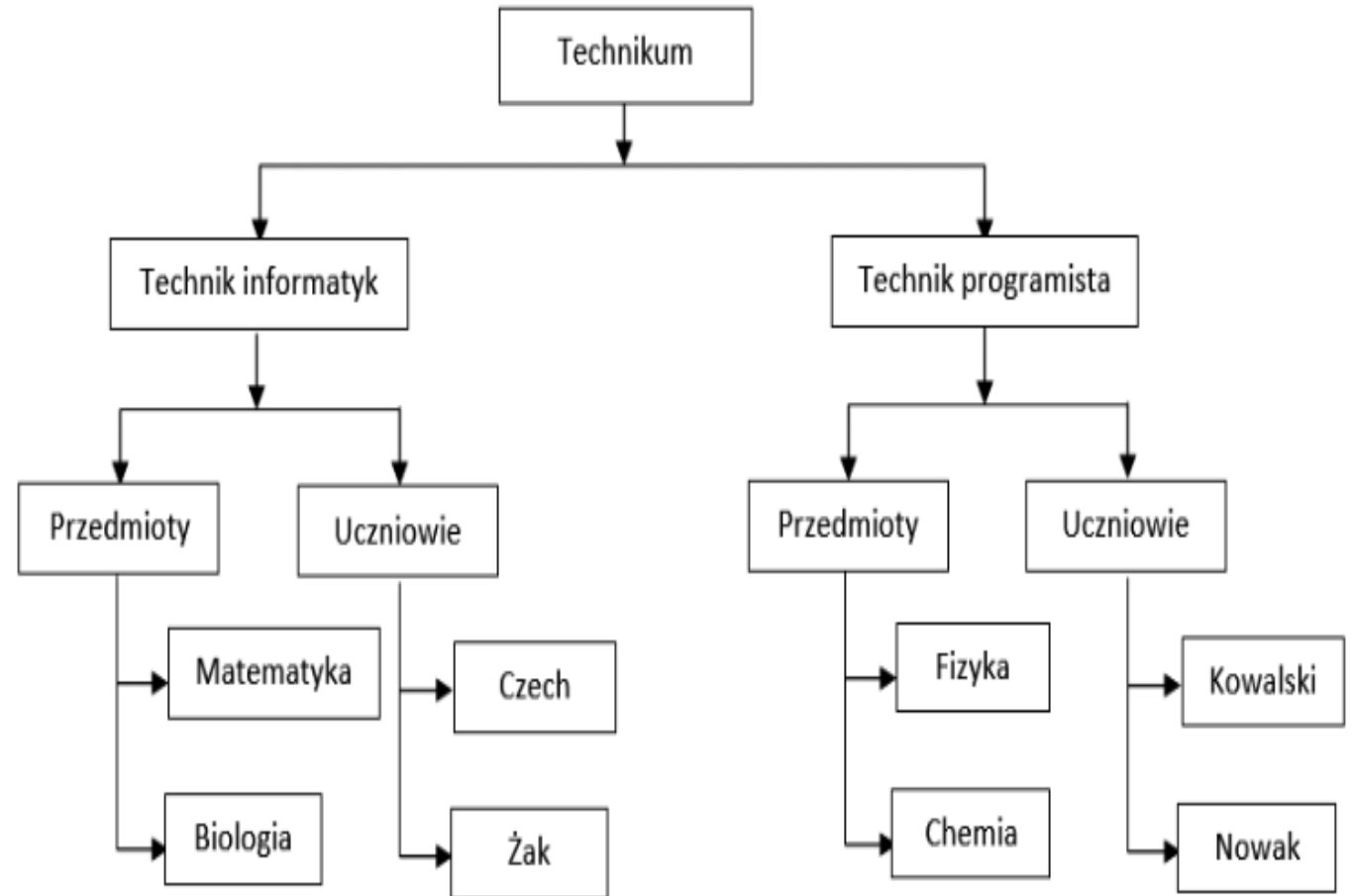
Wszystkie są **relacyjne**.

Wszystkie obsługują SQL

## HIERARCHICZNY

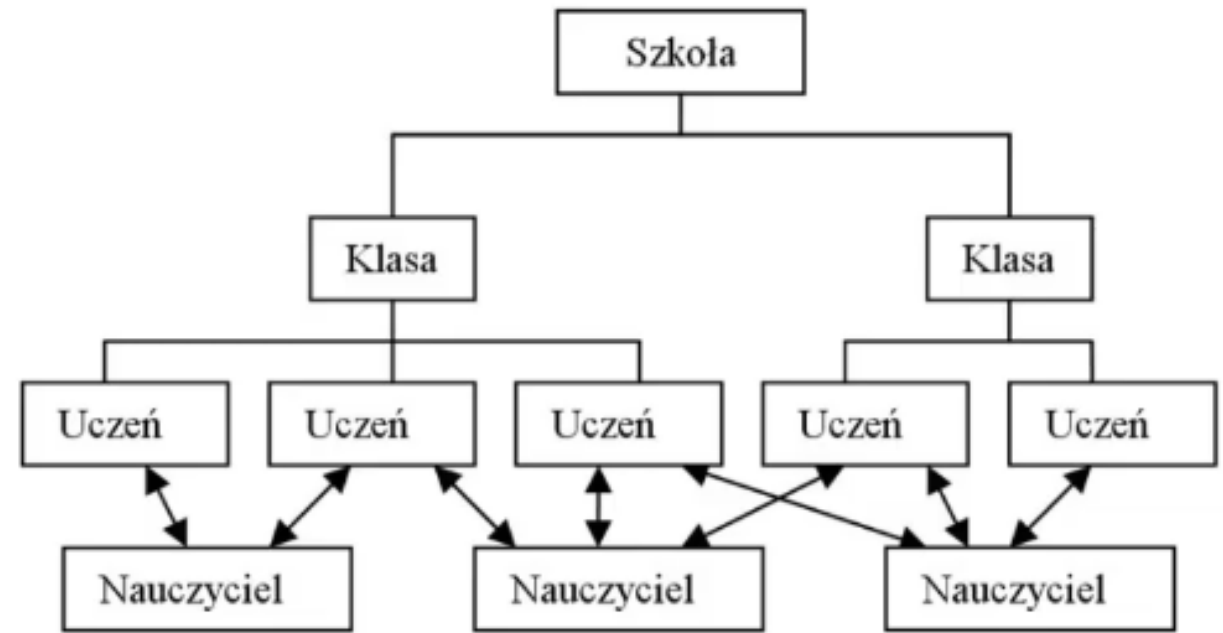
Hierarchiczny model - W tym modelu przechowywane dane są zorganizowane w **postaci odwróconego drzewa**. Informacja jest zawarta w dokumentach oraz w strukturze drzewa

NoSQL, mongoDB opierają się na hierarchiczne modele



## SIECIOWY

Zmodyfikowana wersja modelu hierarchicznego, pozwalająca na definiowanie relacji **wiele-do-wielu** w postaci struktury drzewiastej bez powtarzania poszczególnych wartości w ramach obiektu danych.... Rekordy zawierają pola przechowujące dane.



- Podstawą relacyjnych modeli jest algebra relacji.
- Algebra relacyjna definiuje system operacji na relacjach (tabelach): przecięcie, odejmowanie, łączenie itp.
- Wszystkie te operacje są wyrażone przez SQL (*Structured Query Language*)

**Relacyjna baza danych** – to opisany i zorganizowany zbiór tabel połączonych relacjami – związkami między sobą. Ten sposób przechowywania informacji pozwala na **uniknięcie redundancji** (powtarzania się danych) **oraz przeprowadzanie analiz** na podstawie wielu tabel. Każda tabela składa się z **rekordów** (tak nazywamy pojedyncze wiersze). Poszczególne rekordy składają się z **pól** (komórek), przechowujących jedną daną.



- Relacja – tabela
- Krotka – wiersz
- Klucz podstawowy  
(ang. primary key)

	<b>contact_name</b> character varying (30)	<b>address</b> character varying (60)	<b>city</b> character varying (15)
1	Maria Anders	Obere Str. 57	Berlin
2	Ana Trujillo	Avda. de la Constitución 2222	México D.F.
3	Antonio Moreno	Mataderos 2312	México D.F.
4	Thomas Hardy	120 Hanover Sq.	London
5	Christina Berglund	Berguvsvägen 8	Luleå
6	Hanna Moos	Forsterstr. 57	Mannheim
7	Frédérique Citeaux	24, place Kléber	Strasbourg
8	Martín Sommer	C/ Araquil, 67	Madrid
9	Laurence Leblan	12, rue des Bouchers	Marseille
10	Elizabeth Lincoln	23 Tsawassen Blvd.	Tsawassen
11	Victoria Ashworth	Fauntleroy Circus	London
12	Patricio Simpson	Cerrito 333	Buenos Aires
13	Francisco Chang	Sierras de Granada 9993	México D.F.

**Klucz obcy (ang. foreign key)** – to kolumna w tabeli, która ustanawia powiązanie między danymi w dwóch różnych tabelach. Tworzy relację między tabelami, odwołując się do kolumny (kolumn) klucza podstawowego innej tabeli.

**Customer Table**

Customer ID	Name	Address	Phone#

Primary Key

DatabaseTown.com

**Order Table**

Order No.	Customer ID	Item ID	Order Detail

Foreign Key

- **Relacja jeden do jednego** - oznacza, że dla wiersza w tabeli A może istnieć maksymalnie jeden zgodny wiersz w tabeli B i odwrotnie

PrzewodniczącyKlas			
PrzewodniczacyID	Imie	Nazwisko	Telefon
1	Zbigniew	Pracowity	693456945
2	Andrzej	Sumienny	705763497
3	Paweł	Cichy	623094886

Klasy			
KlasaID	Nazwa	PrzewodniczacyID	LiczbaUczniów
1	I LO	2	30
2	III TI	1	25
3	IV TM	3	27

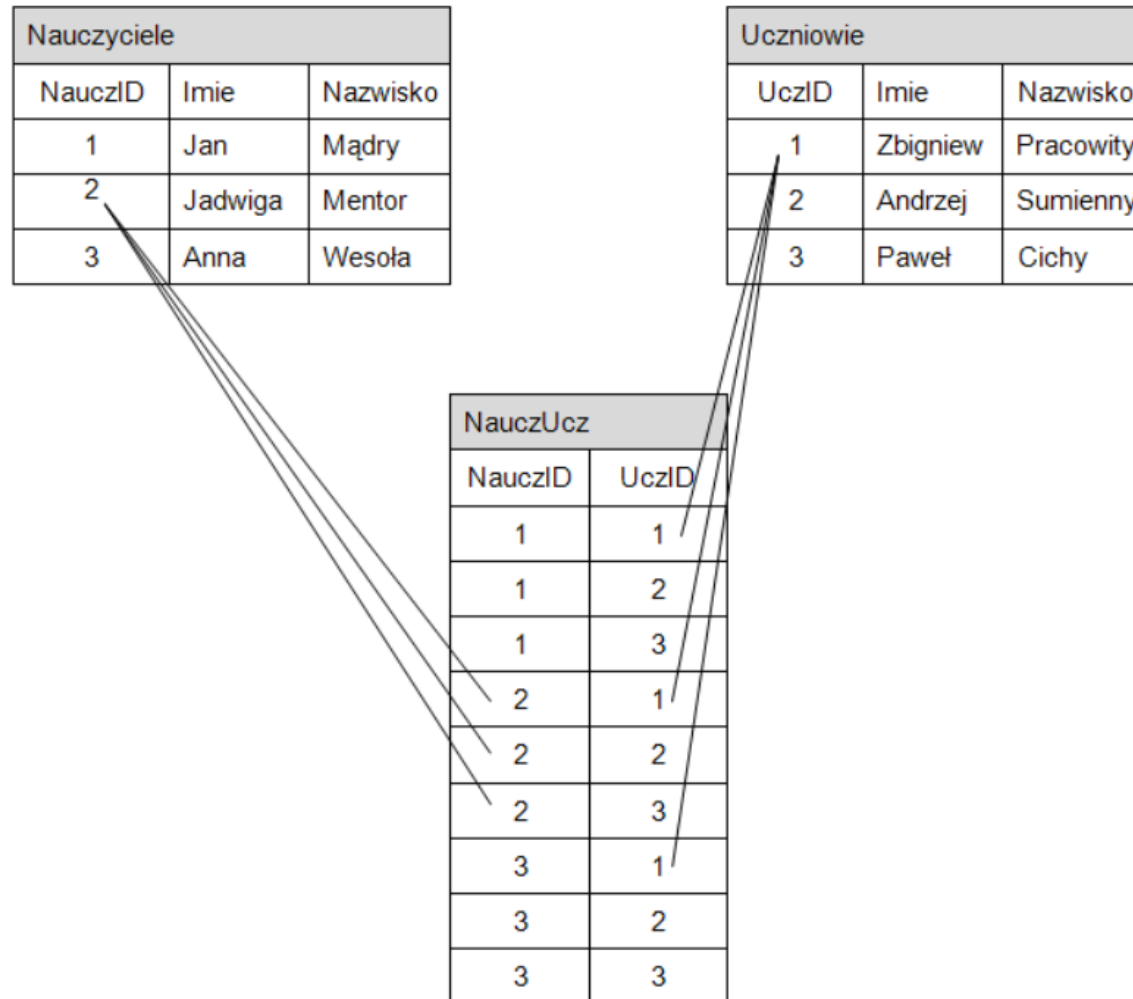
**Relacja jeden do wielu** - jest najbardziej powszechnym rodzajem relacji. W przypadku relacji tego typu dla wiersza w tabeli A może istnieć wiele zgodnych wierszy w tabeli B. Natomiast dla wiersza w tabeli B może istnieć tylko jeden zgodny wiersz w tabeli A.

WychowawcyKlas			
WychowawcaID	Imie	Nazwisko	Telefon
1	Jan	Mądry	693456945
2	Jadwiga	Mentor	705567497
3	Anna	Wesoła	689194886

Klasy			
KlasaID	Nazwa	WychowawcaID	LiczbaUczniów
1	I LO	1	30
2	III TI	1	25
3	IV TM	3	27

**Relacja wiele do wielu** - dla wiersza w tabeli A może istnieć wiele zgodnych wierszy w tabeli B i odwrotnie. Taka relacja jest tworzona przez zdefiniowanie trzeciej tabeli, nazywanej tabelą skrzyżowań. Klucz podstawowy tabeli skrzyżowań składa się z kluczy obcych zarówno z tabeli A, jak i z tabeli B.

## Relacja wiele do wielu i tabela łącząca.



**Normalizacja baz danych** - modyfikacja struktury bazy danych w celu zlikwidowania nadmiarowości danych, oraz ułatwić dostęp do danych. Wyróżniamy 3 podstawowe "standardy" poprawnego tworzenia bazy danych, czyli 3 postacie normalne.

Relacja jest w pierwszej postaci normalnej, jeśli wartości pól są atomowe, czyli zawierają pojedynczą informację.

Tabela nie normalizowana

StudentID	Imię i Nazwisko	Kursy
1	Jan Kowalski	Matematyka, Fizyka
2	Anna Nowak	Biologia, Chemia
3	Piotr Wiśniewski	Informatyka



Relacja jest w drugiej postaci normalnej, wtedy kiedy jest w pierwszej oraz każda tabela powinna przechowywać dane dotyczące tylko konkretnej klasy obiektów.

Tabela nie normalizowana

StudentID	Przedmiot	Ocena	Imię i Nazwisko Studenta	Kierunek Studiów
1	Matematyka	4.0	Jan Kowalski	Informatyka
1	Fizyka	3.5	Jan Kowalski	Informatyka
2	Matematyka	5.0	Anna Nowak	Matematyka

1. Tabela Studenci (przechowuje dane o studentach):

StudentID	Imię	Nazwisko	Kierunek Studiów
1	Jan	Kowalski	Informatyka
2	Anna	Nowak	Matematyka

2. Tabela Oceny (przechowuje dane o ocenach studentów):

StudentID	Przedmiot	Ocena
1	Matematyka	4.0
1	Fizyka	3.5
2	Matematyka	5.0

Relacja jest w trzeciej postaci normalnej, wtedy kiedy jest w drugiej postaci normalnej oraz kolumna informacyjna nie należąca do klucza nie zależy też od innej kolumny informacyjnej, nie należącej do klucza. Każdy niekluczowy argument jest zależny tylko od klucza głównego a nie od innej kolumny.

Tabela nie normalizowana

ZamówienieID	KlientID	KlientNazwa	KlientAdres	Miasto	KodPocztowy
1	1	Kowalski	Ul. Kwiatowa 10	Warszawa	00-001
2	2	Nowak	Ul. Słoneczna 5	Kraków	30-001
3	1	Kowalski	Ul. Kwiatowa 10	Warszawa	00-001

## Problemy w tej tabeli:

- Tabela jest w 2NF, ponieważ dane dotyczące klientów zostały znormalizowane, ale jest problem z zależnością między KlientAdres, Miasto, i KodPocztowy. Miasto i KodPocztowy zależą od KlientAdres, a nie bezpośrednio od klucza głównego (ZamówienieID).

1. Tabela Klienci (przechowuje dane o klientach):

KlientID	KlientNazwa	KlientAdres
1	Kowalski	Ul. Kwiatowa 10
2	Nowak	Ul. Słoneczna 5

2. Tabela Adresy (przechowuje dane o adresach):

KlientAdres	Miasto	KodPocztowy
Ul. Kwiatowa 10	Warszawa	00-001
Ul. Słoneczna 5	Kraków	30-001

3. Tabela Zamówienia (przechowuje dane o zamówieniach):

ZamówienieID	KlientID
1	1
2	2
3	1