

Temat 1 .

dr inż. Michał Malinowski

<https://www.drmalinowski.edu.pl>

Wprowadzenie do baz
danych i model relacyjny

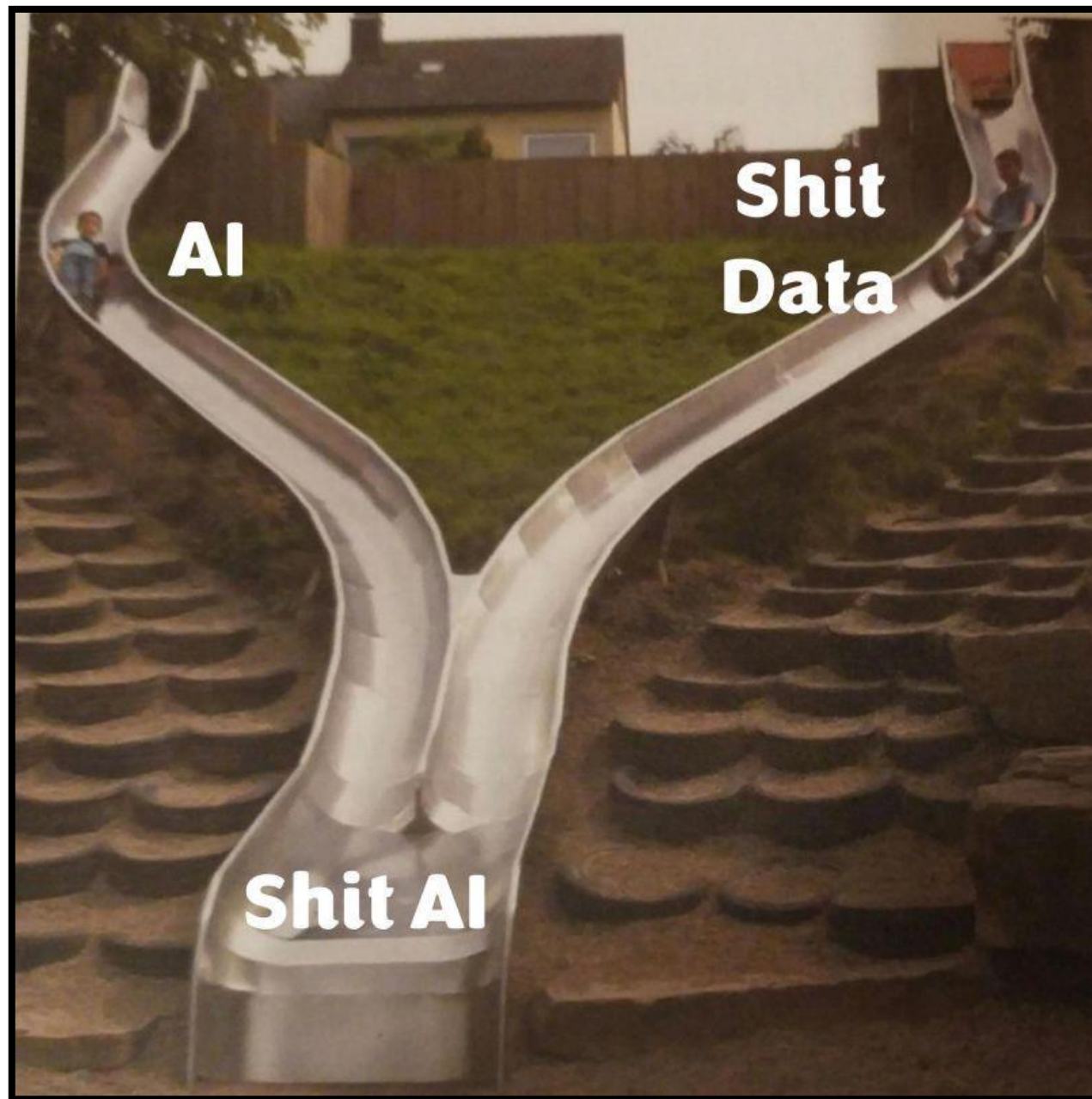
PODSTAWY BAZ DANYCH



- 9



Bazy danych to pokarm dla sztucznej inteligencji.



Definicja i znaczenie baz danych

Baza danych

to zorganizowany zbiór danych, który umożliwia łatwe przechowywanie, zarządzanie, dostęp do informacji oraz dane są zorganizowane w sposób strukturalny.

Biblioteki

- Katalogi kartkowe
- Systemy klasyfikacji

Rejestry publiczne

- Księgi wieczyste
- Akta urodzeń, małżeństw i zgonów

Kroniki i Archiwa

- Kroniki klasztorne
- Archiwa państwowe

Zbiory danych handlowych

- Księgi rachunkowe
- Rejestry magazynowe

Zbiory danych naukowych

- Zielniki Botaniczne
- Tablice astronomiczne

Indeksy w archiwach kościelnych

- Rejestry parafialne

Spisy ludności

- Manuskrypty spisów
- Dane statystyczne

Cyfrowe

- Organizacji i Państwowe
- Prywatne

Definicja i znaczenie baz danych

Rola baz danych w nowoczesnych systemach informatycznych

Centralne repozytorium danych:

- Przechowuje duże ilości danych w jednym miejscu, umożliwiając łatwy dostęp i zarządzanie.

Wspomaganie podejmowania decyzji:

- Bazy danych są kluczowe dla analiz biznesowych, raportowania oraz podejmowania decyzji opartych na danych.

Zwiększenie efektywności operacyjnej:

- Automatyzacja procesów biznesowych i operacji dzięki sprawnemu zarządzaniu danymi.

Ułatwienie integracji systemów:

- Bazy danych umożliwiają integrację różnych aplikacji i systemów poprzez wspólny dostęp do danych.

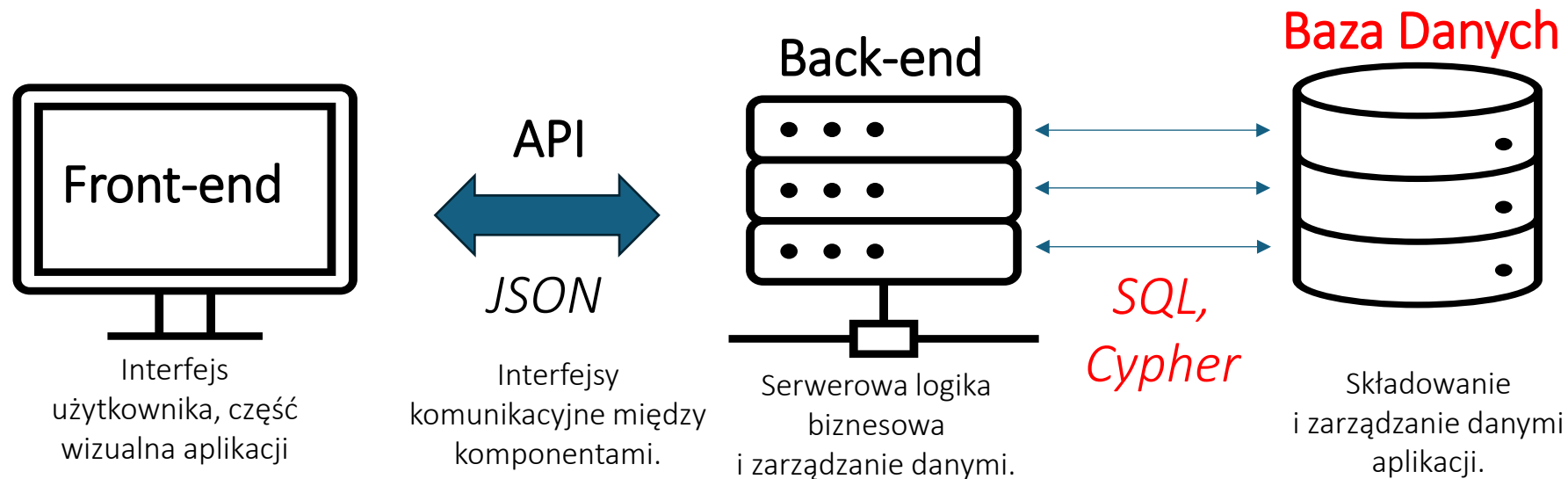
Bezpieczeństwo i ochrona danych:

- Bazy danych oferują mechanizmy zabezpieczające, takie jak kontrola dostępu, szyfrowanie oraz audyt.

Definicja i znaczenie baz danych

Miejsce baz danych w architekturze aplikacji

Składowanie i zarządzanie danymi aplikacji



Historia rozwoju baz danych

Nośniki i pozyskanie informacji na przestrzeni dziejów

Okres	Nośniki informacji	Metoda pozyskania informacji	Szacunkowa wielkość zachowanych danych
Prehistoria	Pamięć plemienna, ściany jaskiń	Pytania do rodziny lub wodza plemienia	Kilka gigabajtów (GB)
Starożytność	Gliniane tabliczki, papirus	Konsultacje z uczonymi lub kapłanami	Dziesiątki gigabajtów (GB)
Średniowiecze	Pergamin, skóra	Manualne przeglądanie skryptoriów w klasztorach	Setki gigabajtów (GB)
Nowożytność	Książki (papier)	Korzystanie z drukowanych katalogów bibliotecznych	Dziesiątki terabajtów (TB)
XIX wiek	Gazety (papier)	Kluby kawiarnie jako miejsca spotkań i dyskusji intelektualistów	Setki terabajtów (TB)
XX wiek	Film i nagrania, pierwsze nośniki cyfrowe	Użycie kartotek. Wyszukiwanie za pomocą internetowych wyszukiwarek	Kilka petabajtów (PB)
XXI wiek	Zaawansowane dyski twarde	Wyszukiwanie za pomocą sztucznej inteligencji	Setki zettabajtów (ZB)

Historia rozwoju baz danych

Historia



Systemy Plikowe

- Dane przechowywane w osobnych plikach na taśmach i dyskach.
- Problemy z duplikacją, integralnością, trudności w zarządzaniu.

1950-1960



Hierarchiczne i Sieciowe Bazy Danych

- Model hierarchiczny: Dane zorganizowane w strukturę drzewa (np. IMS od IBM).
- Model sieciowy: Złożone sieci zależności między danymi (np. CODASYL DBTG).
- Skomplikowane struktury, trudne do modyfikacji.

1960-1970



Relacyjne Bazy Danych

- Teoria relacyjna: Wprowadzona przez Edgara F. Codd, oparta na algebrze relacyjnej.
- Model relacyjny: Dane przechowywane w tabelach, klucze jako relacje między danymi.
- Pojawienie się RDBMS (np. Oracle, IBM DB2, MS SQL Server).

1970-1980



Nowoczesne Relacyjne Bazy Danych

- Wsparcie dla transakcji, integralności danych, zaawansowane indeksowanie.
- Rozwój standardu SQL (procedury składowane, wyzwalacze, widoki).
- Wprowadzenie OLAP, hurtowni danych, migracja do chmury (np. Amazon RDS).

1980-obecnie



Alternatywne Modele Danych i NoSQL

- NoSQL: Elastyczne rozwiązania dla dużych zbiorów danych (np. MongoDB, Cassandra).
- Bazy grafowe: Przetwarzanie złożonych relacji (np. Neo4j).
- Przewidywany rozwój w kierunku hybrydowych modeli i integracji z AI.

2000-obecnie

Historia rozwoju baz danych

Współczesne Bazy Danych

Typ Bazy Danych	Nazwa	Opis
Relacyjne	MySQL	Open-source system zarządzania bazą danych, popularny w aplikacjach webowych.
	Oracle Database	Jeden z najbardziej zaawansowanych systemów baz danych, używany w dużych aplikacjach korporacyjnych.
	Microsoft SQL Server	System zarządzania bazą danych opracowany przez Microsoft, szeroko stosowany w korporacjach.
	PostgreSQL	Zaawansowany system zarządzania bazą danych, znany z obsługi skomplikowanych zapytań SQL.
	SQLite	Lekka, wbudowana baza danych, idealna do aplikacji mobilnych i desktopowych.
Nierelacyjne (NoSQL)	Neo4j	Baza danych grafowa, która przechowuje dane w postaci węzłów, relacji i właściwości.
	MongoDB	Baza danych dokumentowa, która przechowuje dane w formacie JSON.
	Cassandra	Baza danych kolumnowa, zaprojektowana do obsługi dużych ilości danych rozproszonych.
	Redis	Baza danych klucz-wartość, znana z wysokiej wydajności i wsparcia dla różnych struktur danych.
	CouchDB	Baza danych dokumentowa, która przechowuje dane w formacie JSON z obsługą wersjonowania.

Cykl DIKW

Dane (Data)

są surowymi, nieprzetworzonymi faktami, które nie mają przypisanego znaczenia ani kontekstu.

Są to liczby, teksty, obrazy czy dźwięki, które same w sobie nie dostarczają wiedzy. Przykładami danych mogą być pojedyncze liczby (np. 100), słowa (np. „czerwony”) lub symbole

Informacja (Information)

to dane, które zostały zinterpretowane i przypisano im znaczenie w określonym kontekście.

Kiedy dane zostaną przetworzone, połączone i zorganizowane, mogą dostarczyć odpowiedzi na pytania, co, kto, kiedy i gdzie. Informacja ma wartość, ponieważ pozwala na zrozumienie kontekstu, w którym funkcjonują dane. Przykład informacji: „100 zł zostało wydane na zakupy w sklepie spożywczym w poniedziałek.”

Cykl DIKW

Wiedza (Knowledge)

to połączenie informacji z doświadczeniem, intuicją, regułami oraz teoriami.

Wiedza to zdolność do rozumienia i analizowania informacji, co prowadzi do podjęcia decyzji i działań. Jest to zasób, który pomaga w rozwiązywaniu problemów i podejmowaniu racjonalnych decyzji. Przykład wiedzy: „Na podstawie wydatków w sklepie spożywczym można wywnioskować, że wzrosły ceny produktów.”



Cykl DIKW

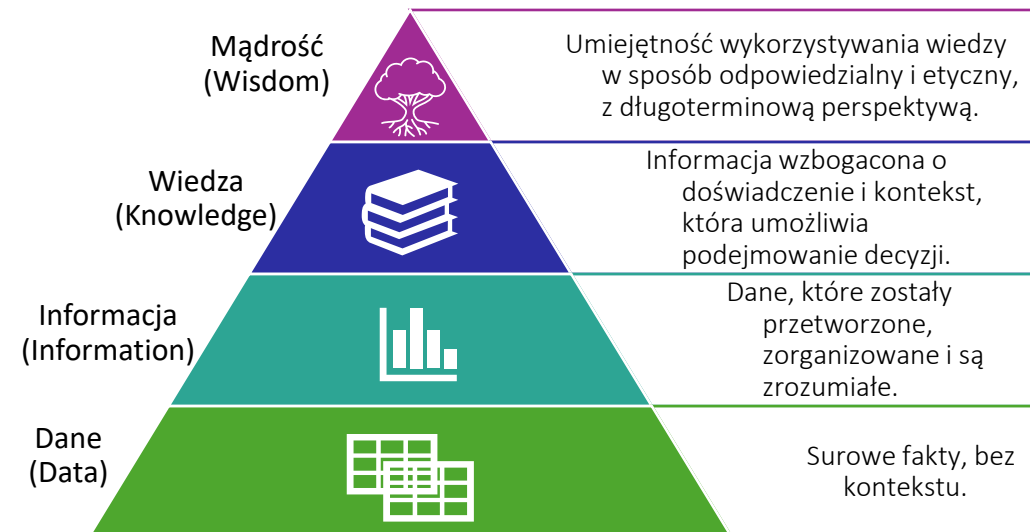
Mądrość (Wisdom)

oznacza zdolność do korzystania z wiedzy w sposób, który przynosi korzyści długoterminowe oraz uwzględnia złożoność rzeczywistości. W biznesie mądrość oznacza podejmowanie strategicznych decyzji, które uwzględniają zarówno cele krótko-, jak i długoterminowe, oraz szerszy kontekst społeczny, środowiskowy i etyczny.

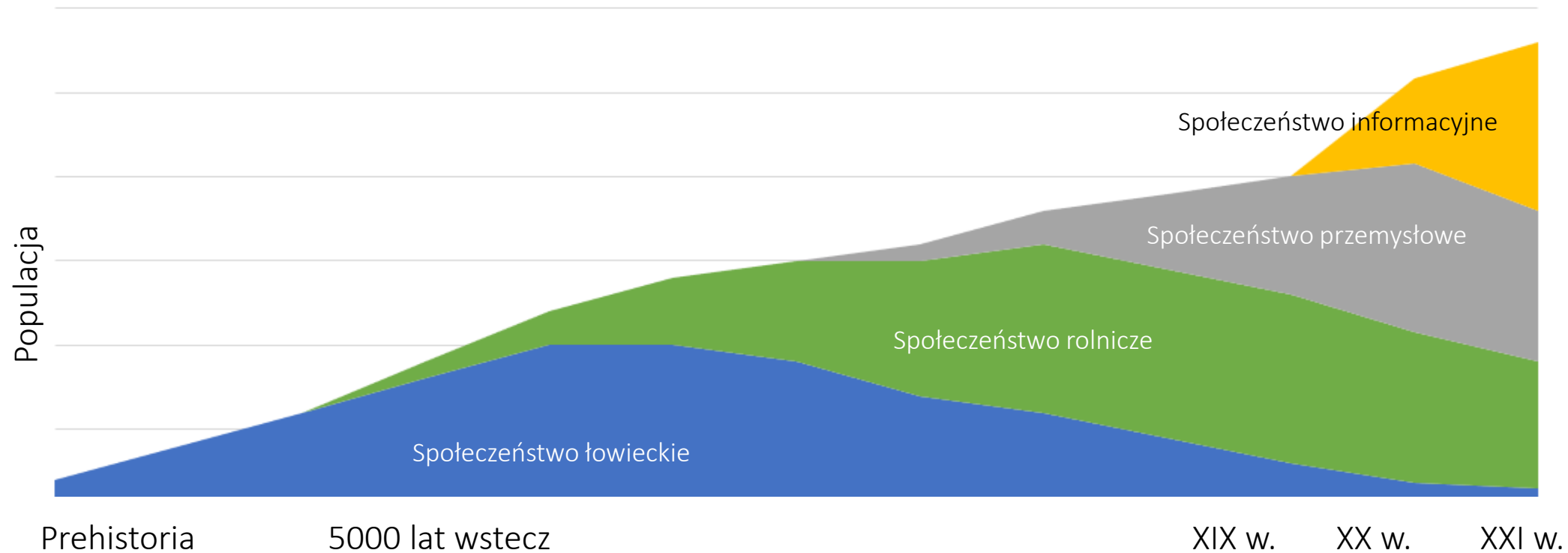
Jest najwyższym poziomem w tej hierarchii.

Cykl DIKW (Dane, Informacja, Wiedza, Mądrość)

przedstawia hierarchię przekształcania danych w coraz bardziej wartościowe formy.



Gdzie teraz jesteśmy... ?



Informacja - najważniejszy zasób

Społeczeństwo informacyjne

podstawą społeczeństwa informacyjnego
jest gospodarka oparta na wiedzy,
a jej zasadniczym zasobem gospodarczym
są informacje skumulowane w bazach.



Kreacja informacji

Tworzenie danych
Zbieranie danych
Analiza danych
Udostępnianie danych
Konsolidacja danych

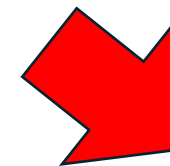
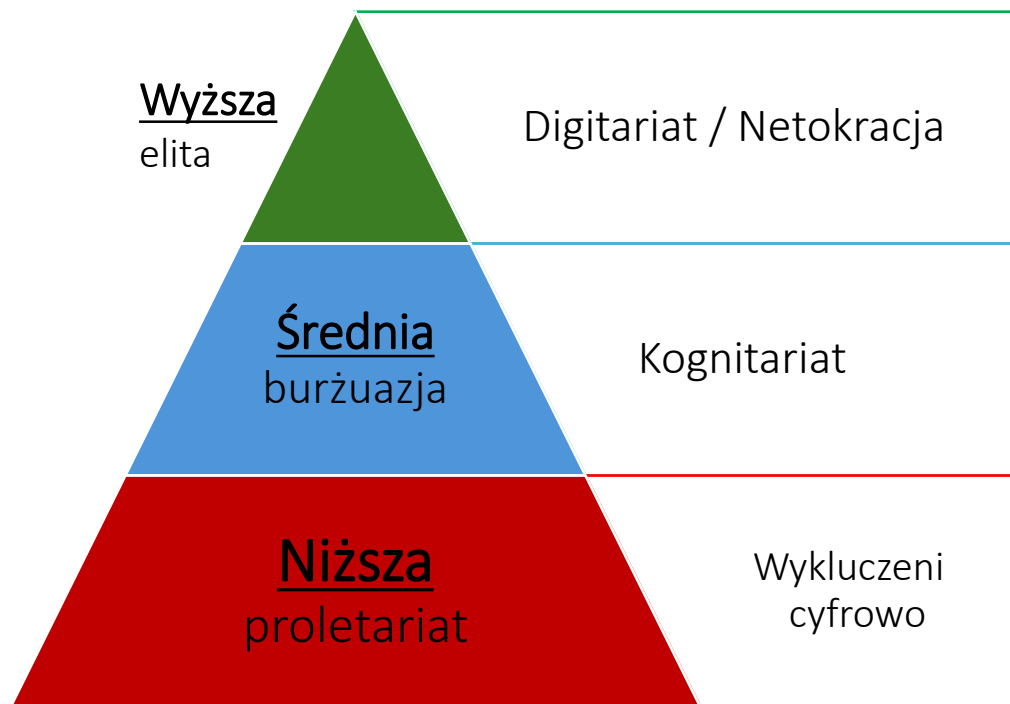
Zbiory danych

Autor

Czytelnik



Klasy społeczne



Destrukcja informacji

Usuwanie danych
Dezinformacja
Zniekształcenie danych
Brak aktualizacji
Zaniedbania w archiwizacji

Ataki cybernetyczne

Obrońca

Atakujący



Modele baz danych

SQL (70%)

Bazy relacyjne *

Przechowują dane w tabelach powiązanych kluczami głównymi i obcymi, zapewniając integralność i spójność danych.

Przykłady: Oracle, MSSQL, MySQL

Zastosowanie: Systemy transakcyjne, raportowanie, analizy danych

NoSQL (20%)

Bazy dokumentowe

Przechowują dane w formie dokumentów, zazwyczaj w formacie JSON, BSON lub XML.

Przykłady: MongoDB, CouchDB.

Zastosowanie: Idealne do przechowywania danych o zmiennej strukturze, takich jak dokumenty, profile użytkowników.

Bazy klucz-wartość

Przechowują dane jako pary klucz-wartość, gdzie klucz jest unikalnym identyfikatorem, a wartość może być dowolnym typem danych.

Przykłady: Redis, DynamoDB.

Zastosowanie: Świetne do szybkiego przechowywania i pobierania dużych ilości danych, takich jak sesje użytkowników, dane cache.

Bazy kolumnowe

Organizują dane w kolumnach zamiast w wierszach, co umożliwia efektywne przechowywanie i przetwarzanie dużych zbiorów danych analitycznych.

Przykłady: Apache Cassandra, HBase.

Zastosowanie: Idealne do analizy dużych zestawów danych, gdzie istotne są operacje na kolumnach danych.

Bazy grafowe *

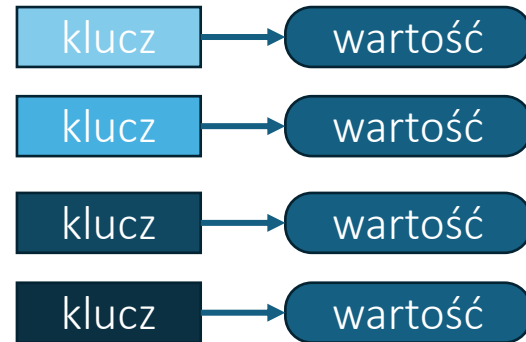
Przechowują dane w postaci węzłów, krawędzi i właściwości, co umożliwia modelowanie złożonych relacji między danymi.

Przykłady: Neo4j, ArangoDB.

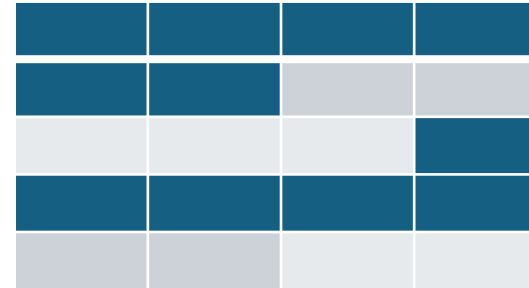
Zastosowanie: Używane w aplikacjach wymagających analizy grafów, takich jak sieci społecznościowe, zarządzanie zasobami, rekomendacje.

Modele baz danych

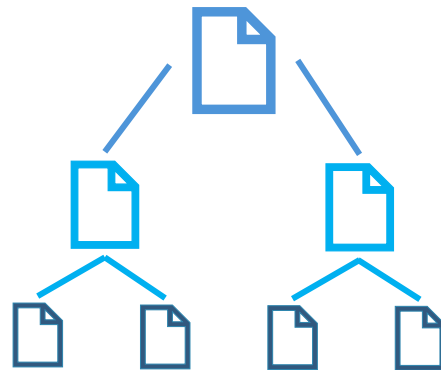
Bazy par klucz-wartość



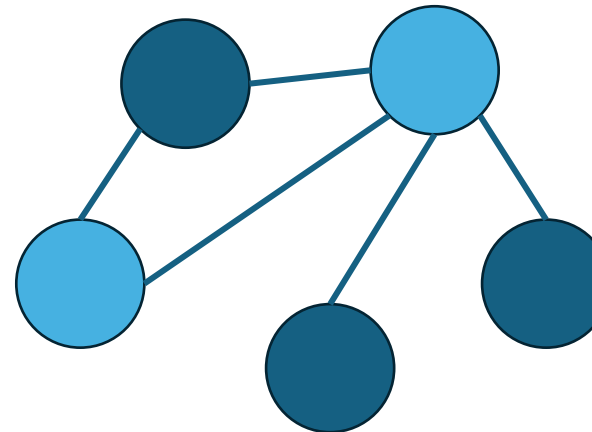
Bazy kolumnowe



Bazy dokumentów



Bazy grafowe



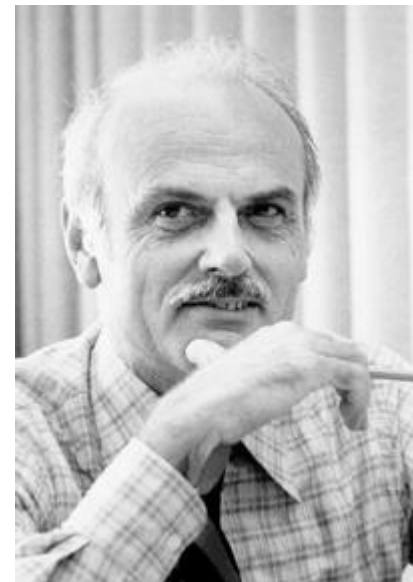
Model relacyjny

Edgar Frank Codd

Edgar Frank „Ted” Codd (ur. 23 sierpnia 1923 w Isle of Portland, zm. 18 kwietnia 2003 w Williams Island, Aventura) – brytyjski informatyk, wniósł wkład w rozwój teorii relacyjnych baz danych.

W 1970 roku wydał fundamentalną pracę ***A Relational Model of Data for Large Shared Data Banks***, w której przedstawił relacyjny model zarządzania bazami danych.

Ku jego rozczarowaniu IBM (gdzie pracował) zwlekał z wdrożeniem zaprezentowanych rozwiązań aż do chwili, gdy konkurencyjne firmy zaczęły stosować ten model – przykładem była baza danych Oracle, opracowana przez Larry’ego Ellisona na bazie idei Codda



Model relacyjny

Tabele (Tables)

Dane w bazach relacyjnych są przechowywane w tabelach, które składają się z wierszy (rekordów) i kolumn (atrybutów). Każda tabela reprezentuje zbiór podobnych obiektów, np. klientów, produktów czy zamówień.

Rekordy (Records)

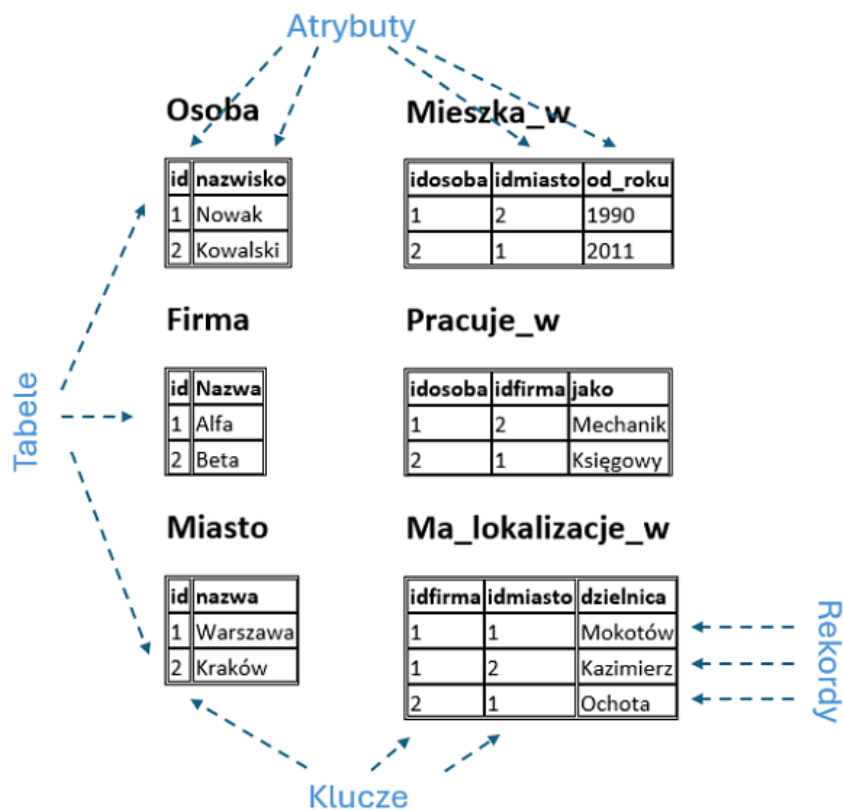
Każdy wiersz w tabeli nazywany jest rekordem i reprezentuje pojedynczy obiekt lub jednostkę danych. Na przykład, w tabeli "Klienci" każdy rekord może odpowiadać jednemu klientowi.

Atrybuty (Attributes)

Kolumny w tabeli określają atrybuty obiektów przechowywanych w tabeli. Na przykład, tabela "Klienci" może mieć kolumny takie jak "ID", "Imię", "Nazwisko" i "Adres".

Model relacyjny

Elementy



Relacje (Relationships)

Bazy relacyjne umożliwiają definiowanie relacji między tabelami poprzez klucze.

Relacje te mogą być typu jeden-do-jednego, jeden-do-wielu lub wiele-do-wielu, co pozwala na powiązanie danych z różnych tabel.

Klucze (Keys)

- Klucz główny (Primary Key) to unikalny identyfikator każdego rekordu w tabeli.
- Klucze obce (Foreign Keys) są używane do tworzenia relacji między tabelami, umożliwiając łączenie danych z różnych tabel.

Integralność i transakcja

Integralność

Zasada zapewniająca poprawność, spójność i wiarygodność danych w bazie danych poprzez restrykcje takie jak klucze główne, obce, oraz zasady integralności referencyjnej i domenowej.

Integralność Encji:

Zapewnia, że każda encja (rekord) w tabeli bazy danych jest unikalna i możliwa do zidentyfikowania.

Integralność Referencyjna:

Klucz główny: Każda tabela musi mieć klucz główny (Primary Key), który jednoznacznie identyfikuje każdy wiersz w tabeli. Klucz główny nie może zawierać wartości NULL i musi być unikalny.

Zapewnia spójność między powiązаныmi tabelami poprzez zachowanie poprawnych relacji między kluczami głównymi a kluczami obcymi.

Klucz obcy: Klucz obcy (Foreign Key) w jednej tabeli musi odpowiadać wartości istniejącego klucza głównego w innej tabeli, co zapobiega powstaniu "osieroconych" rekordów.

Integralność Domenowa:

Zapewnia, że dane w kolumnach bazy danych są zgodne z określonymi regułami (domeną), czyli wartościami dopuszczalnymi dla danej kolumny.

Ograniczenia domenowe: Typ danych, zakres wartości, wzorce (np. wyrażenia regularne) oraz inne zasady ograniczające poprawność i zgodność danych.

Integralność i transakcja

Transakcja

to sekwencja operacji na bazie danych, traktowana jako jedna, niepodzielna jednostka pracy. Transakcja musi być wykonana w całości (commit) lub w ogóle (rollback), aby zapewnić spójność danych oraz musi spełniać właściwości ACID:

Atomicity (Atomowość):

- Wszystkie operacje w ramach transakcji są wykonane w całości lub żadna z nich. Jeśli jedna operacja się nie powiedzie, cała transakcja jest cofana.

Consistency (Spójność):

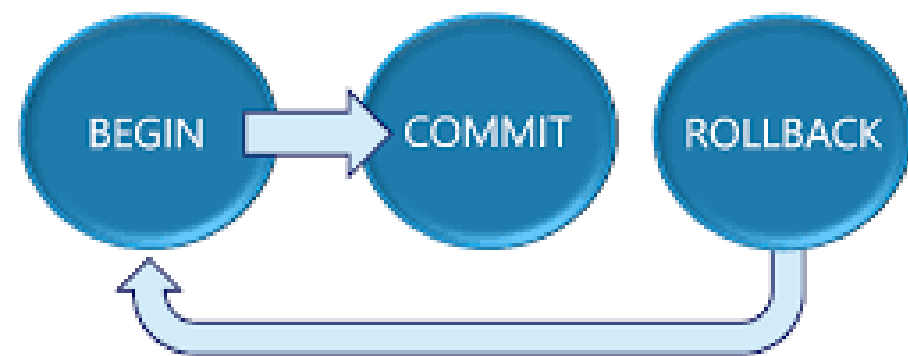
- Transakcja przekształca bazę danych z jednego poprawnego stanu do innego, zachowując przy tym wszystkie reguły integralności.

Isolation (Izolacja):

- Wyniki transakcji są niewidoczne dla innych transakcji, dopóki nie zostaną zatwierdzone, co zapobiega zakłóceniom między współbieżnymi transakcjami.

Durability (Trwałość):

- Po zatwierdzeniu transakcji, jej wyniki są trwale zapisane w bazie danych, nawet w przypadku awarii systemu.



Operacje na relacjach

Operacja	Opis	Przykład
Selekcja (SELECT)	Wybiera wiersze z tabeli, które spełniają określone warunki.	Wybranie wszystkich klientów z miasta Warszawa.
Projekcja (PROJECT)	Wybiera określone kolumny z tabeli, eliminując duplikaty wierszy.	Wyświetlenie tylko nazwisk i adresów klientów.
Łączenie (JOIN)	Łączy dane z dwóch lub więcej tabel na podstawie wspólnego atrybutu.	Powiązanie danych o zamówieniach z danymi o klientach.
Różnica (DIFFERENCE)	Zwraca różnicę między dwiema tabelami, wybierając wiersze z jednej tabeli, których nie ma w drugiej.	Znalezienie klientów, którzy nie złożyli żadnego zamówienia.
Suma (UNION)	Łączy dane z dwóch tabel, zwracając wszystkie wiersze, eliminując duplikaty.	Połączenie listy klientów z różnych oddziałów w jedną listę.
Iloczyn Kartezjański (CARTESIAN PRODUCT)	Tworzy wszystkie możliwe kombinacje wierszy z dwóch tabel.	Połączenie każdej kombinacji produktów z każdym zamówieniem (zwykle z filtrem).

THE FOUR HORSEMEN OF PROCRASTINATION

