

# Zadania z ERD i normalizacji

Relacyjne Bazy Danych · projektowanie i postacie normalne

Modele zapisujemy w postaci **schematów relacji**: TABELA(klucz\_główny, atrybut, atrybut\_FK → TABELA2). Klucz główny podkreślony, klucz obcy oznaczony „→”. To dokładnie ten zapis, który powstaje z diagramu ERD i jest oceniany na egzaminie.

## Część I — Modelowanie ERD

### Zadanie 1 · Rozpoznawanie typu związku

Określ typ związku (1:1, 1:N czy M:N) w każdej sytuacji:

- Dział zatrudnia wielu pracowników; pracownik pracuje w jednym dziale.
- Student może zapisać się na wiele kursów; na kurs zapisuje się wielu studentów.
- Każdy pracownik ma dokładnie jedną kartę dostępu; karta należy do jednego pracownika.
- Pracownik ma jednego szefa; szef kieruje wieloma pracownikami.

#### Rozwiązanie

a) **1:N** (Dział → Pracownicy) · b) **M:N** (Student ↔ Kurs) · c) **1:1** (Pracownik ↔ Karta) · d) **1:N rekurencyjny** (Pracownik → Pracownik).

### Zadanie 2 · Model biblioteki z wymagań

Zamodeluj bazę biblioteki: biblioteka przechowuje **książki** (tytuł, ISBN, autor); każdej książki może być wiele **egzemplarzy** (sygnatura, stan). **Czytelnicy** (imię, nazwisko, nr karty) **wypożyczają** egzemplarze (data wypożyczenia, data zwrotu). Podaj schematy relacji.

#### Rozwiązanie

```
KSIAZKI(id_ksiazki PK, tytuł, isbn, autor)
EGZEMPLARZE(id_egz PK, sygnatura, stan,
id_ksiazki → KSIAZKI)
CZYTELNICY(id_czyt PK, imię, nazwisko, nr_karty)
WYPOZYCZENIA(id_wyp PK, data_wyp, data_zwrotu,
id_egz → EGZEMPLARZE,
id_czyt → CZYTELNICY)
```

Książka 1—N Egzemplarz (klucz obcy po stronie egzemplarza). Wypożyczenie łączy egzemplarz i czytelnika — realizuje związek M:N „czytelnik wypożycza egzemplarze” z atrybutami dat.

### Zadanie 3 · Rozwiązanie związku M:N

Studenci zapisują się na kursy (związek M:N), a przy zapisie zapamiętujemy ocenę. Pokaż, jak odwzorować to w tabelach.

#### Rozwiązanie

```
STUDENCI(id_stud PK, imię, nazwisko)
KURSY(id_kursu PK, nazwa, ects)
ZAPISY(id_stud → STUDENCI,
id_kursu → KURSY,
ocena,
PRIMARY KEY (id_stud, id_kursu))
```

Związek M:N wymaga **tabeli łączącej (asocjacyjnej)**. Jej kluczem głównym jest para kluczy obcych. Atrybut związku (ocena) trafia właśnie do tej tabeli.

#### Zadanie 4 · Związek kategorii (generalizacja)

Na uczelni są **osoby** (PESEL, imię, nazwisko). Osoba jest **studentem** (nr albumu, rok) albo **wykładowcą** (tytuł, pensja). Zaproponuj odwzorowanie z osobną tabelą nadtypu i podtypów.

#### Rozwiązanie

```
OSOBY(id_osoby PK, pesel, imie, nazwisko, typ)
STUDENCI(id_osoby PK → OSOBY, nr_albumu, rok)
WYKLADOWCY(id_osoby PK → OSOBY, tytuł, pensja)
```

Wariant „nadtyp + podtypy”: wspólne atrybuty w OSOBY; tabele podtypów dzielą ten sam klucz (jednocześnie PK i FK do OSOBY). Kolumna *typ* to dyskryminator. Alternatywy: jedna tabela ze wszystkimi atrybutami (+ wiele NULL) lub osobne tabele bez nadtypu.

## Część II — Normalizacja

### Zadanie 5 · Zależności funkcyjne i klucz

Dana jest relacja ZAMOWIENIE(nr\_zam, nr\_klienta, nazwa\_klienta, nr\_produktu, nazwa\_produktu, ilosc) z zależnościami: nr\_zam → nr\_klienta; nr\_klienta → nazwa\_klienta; nr\_produktu → nazwa\_produktu; (nr\_zam, nr\_produktu) → ilosc. Wskaż klucz główny.

#### Rozwiązanie

Kluczem jest **(nr\_zam, nr\_produktu)** — tylko ta para funkcyjnie wyznacza wszystkie pozostałe atrybuty (przez łańcuch zależności). Pojedynczo żaden atrybut nie identyfikuje wiersza, bo jedno zamówienie ma wiele produktów, a jeden produkt występuje w wielu zamówieniach.

### Zadanie 6 · Doprowadzenie do 1NF

Tabela FAKTURA przechowuje w jednej komórce listę pozycji: (nr\_faktury, klient, pozycje='mleko:2, chleb:3'). Dlaczego narusza 1NF i jak to naprawić?

#### Rozwiązanie

Narusza 1NF, bo kolumna *pozycje* nie jest **atomowa** (lista wartości). Naprawa — wydzielić *pozycje* do osobnej tabeli, jeden wiersz na pozycję:

```
FAKTURY(nr_faktury PK, klient)
POZYCJE(nr_faktury → FAKTURY,
produkt, ilosc,
PRIMARY KEY (nr_faktury, produkt))
```

### Zadanie 7 · Zależność częściowa → 2NF

Relacja jest w 1NF: ZAPISY(nr\_studenta, nr\_kursu, nazwa\_kursu, ocena), gdzie nr\_kursu → nazwa\_kursu. Czy jest w 2NF? Jeśli nie — rozbij ją.

#### Rozwiązanie

**Nie** — występuje **zależność częściowa**: nazwa\_kursu zależy tylko od części klucza (nr\_kursu), a nie od całego klucza złożonego. Rozkład do 2NF:

```
KURSY(nr_kursu PK, nazwa_kursu)
ZAPISY(nr_studenta → STUDENCI,
nr_kursu → KURSY,
ocena,
PRIMARY KEY (nr_studenta, nr_kursu))
```

### Zadanie 8 · Zależność przechodnia → 3NF

Relacja PRACOWNIK(id\_prac, nazwisko, id\_dzialu, nazwa\_dzialu, lokalizacja\_dzialu), gdzie id\_dzialu → nazwa\_dzialu, lokalizacja\_dzialu. Czy jest w 3NF? Rozbij ją.

#### Rozwiązanie

**Nie** — atrybuty działu zależą od id\_dzialu, który nie jest kluczem → **zależność przechodnia** (id\_prac → id\_dzialu → nazwa\_dzialu). Rozkład do 3NF:

```
DZIALY(id_dzialu PK, nazwa_dzialu, lokalizacja_dzialu)
PRACOWNICY(id_prac PK, nazwisko,
id_dzialu → DZIALY)
```

To usuwa redundancję (nazwa i lokalizacja działu nie powtarzają się przy każdym pracowniku) i anomalie aktualizacji/usuwania.

### Zadanie 9 · Rozpoznanie postaci normalnej

W której postaci normalnej jest relacja PROJEKT(id\_proj, nazwa, id\_kierownika, nazwisko\_kierownika), gdzie id\_kierownika → nazwisko\_kierownika?

#### Rozwiązanie

W **2NF**, ale **nie w 3NF**. Klucz jest jednoatrybutowy, więc nie ma zależności częściowych (2NF spełnione), ale istnieje **zależność przechodnia** id\_proj → id\_kierownika → nazwisko\_kierownika, co łamie 3NF.

### Zadanie 10 · Skrót definicji postaci normalnych

Uzupełnij, co eliminuje każda postać normalna (przydatne na egzaminie):

#### Rozwiązanie

**1NF** — wartości atomowe (brak list/grup powtarzalnych). **2NF** — brak zależności częściowych (od części klucza). **3NF** — brak zależności przechodnich (niekluczowy → niekluczowy). **BCNF** — każdy wyznacznik jest nadkluczem. **4NF** — brak zależności wielowartościowych. **5NF** — brak zależności złączeniowych.