Wykład 3
Normalizacja
6. Normalizacja

**TEMATYKA:**
- Cel normalizacji
- Etapy normalizacji
- Związki zależności funkcyjne i niefunkcyjne
- Postacie normalne
- Diagramy zależności
- Akomodacja – przekształcanie diagramu zależności w schemat relacyjny
Opis modelu relacyjnego w notacji nawiązowej

Metoda pozwala w postaci skróconej opisać definicję schematu bazy.

W opisie najpierw podajemy nazwę relacji (tabeli) a następnie w nawiązie kolejne atrybuty. Jako pierwsze podaję się atrybuty stanowiące klucz główny jednocześnie podkreślając je.

Dla przykładu opisanego wcześniej zapis będzie miał postać:

Moduły(NazwaModułu, NrPrac)
Wykładowcy(NrPrac, Pracownik)
Oceny(NazwaModułu, NrStud, TypOceny, Ocena)
Studenci(NrStudenta, Student)
6.1. Cel normalizacji

[*] Normalizacja to proces upraszczania struktury bazy danych w taki sposób, aby osiągnęła ona postać optymalną.

[*] Normalizacja wykonuje się na etapie projektowania modelu fizycznego danych

[*] Dzięki normalizacji można uniknąć anomalii – błędów lub niespójności w bazie danych (w tym również redundancji).

[*] Można wyróżnić 3 rodzaje anomalii:
   - anomalia przy wstawianiu rekordu
     - dopisanie rekordu powoduje dezaktualizacje innego pala,
   - anomalia przy usuwaniu rekordu
     - usunięcie wiersza powoduje usunięcie większej ilości informacji niż żeśmy zamierzali,
   - anomalia przy modyfikacji rekordu
     - zmiana jednego rekordu powoduje konieczność zmiany zapisów w innych rekordach.
Przykład

Dysponujemy bazą danych z informacjami o studentach, modułach oraz wykładowcach na uniwersytecie

<table>
<thead>
<tr>
<th>Nazwa Modułu</th>
<th>NrPrac</th>
<th>Pracownik</th>
<th>NrStud</th>
<th>Student</th>
<th>Ocena</th>
<th>Typ Oceny</th>
</tr>
</thead>
<tbody>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>244</td>
<td>Buczek Jan</td>
<td>34698</td>
<td>Kowalski H.</td>
<td>4.0</td>
<td>zal.</td>
</tr>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>244</td>
<td>Buczek Jan</td>
<td>34698</td>
<td>Kowalski H.</td>
<td>3.5</td>
<td>egz.</td>
</tr>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>244</td>
<td>Buczek Jan</td>
<td>37653</td>
<td>Nowak R.</td>
<td>3.0</td>
<td>zal.</td>
</tr>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>244</td>
<td>Buczek Jan</td>
<td>34610</td>
<td>Lech M.</td>
<td>5.0</td>
<td>zal.</td>
</tr>
<tr>
<td>Projektowanie relacyjnych baz danych</td>
<td>244</td>
<td>Buczek Jan</td>
<td>34698</td>
<td>Kowalski H.</td>
<td>3.0</td>
<td>zal.</td>
</tr>
<tr>
<td>Projektowanie relacyjnych baz danych</td>
<td>244</td>
<td>Buczek Jan</td>
<td>34698</td>
<td>Kowalski H.</td>
<td>4.0</td>
<td>egz.</td>
</tr>
<tr>
<td>Obiektowe bazy danych</td>
<td>445</td>
<td>Kalita Henryk</td>
<td>35785</td>
<td>Woś S.</td>
<td>3.5</td>
<td>egz.</td>
</tr>
<tr>
<td>Roproszone bazy danych</td>
<td>247</td>
<td>Wysocki Edward</td>
<td>34789</td>
<td>Janda K.</td>
<td>5.0</td>
<td>zal.</td>
</tr>
</tbody>
</table>

- Co będzie gdy usuniemy studenta Wosia? – stracimy informację o Obiektowych bazach danych i wykładowcy Kalicie Henryku – anomalia przy usuwaniu
- Co będzie gdy zmienimy wykładowcę Rozproszonych baz danych? – musimy zmodyfikować dwa pola: NrPrac oraz Pracownik – anomalia przy modyfikacji
- Co będzie gdy wpiszemy nowego studenta na moduł? – możliwe to będzie dopiero po uzyskaniu przez niego pierwszego zaliczenia – anomalia przy wstawianiu
6.2. Etapy normalizacji

1. Zebranie danych
2. Przekształcenie do pierwszej postaci normalnej (1PN)
3. Przekształcenie do drugiej postaci normalnej (2PN)
4. Przekształcenie do trzeciej postaci normalnej (3PN)

Po znormalizowaniu do 3 PN najczęściej tablice są już pozbawione anomalii, jeżeli nie to należy je:

- Przekształcić do postaci normalnej Boyce’a-Codd’a (BCNF)
- Przekształcić do czwartej postaci normalnej (4PN)
- Przekształcić do piątej postaci normalnej (5PN)

Proces normalizacji jest włożony. To znaczy, że każda wyższa postać normalna jest podzbiorem postaci niższej.
6.2.1 Rozkład relacji (tablic) a normalizacja

Proces przekształcania nieznormalizowanego zbioru danych w pełni znormalizowaną bazę danych nosi nazwę dekompozycji odwracalnej (rozkładu odwracalnego)

🌟 Cechy dekompozycji odwracalnej

- usuwa redundancję z relacji
- można ją odwrócić przez naturalne złączenie
- powinna doprowadzić relacje do tzw. postaci normalnej
- nie powinna powodować utraty zależności istniejących w relacji pierwotnej
6.3. Zależności funkcyjne i niefunkcyjne (wielowartościowe)

- Dwa elementy A i B są w związku zależności (związku determinowania), jeżeli pewne wartości elementu danych B występują zawsze z pewnymi wartościami elementu A.

- Zależność funkcyjna (determinowanie) między elementami danych wskazuje kierunek związku. Jeżeli A determinuje B to związek jest od A do B nie odwrotnie.

Element danych B jest funkcyjnie zależny od elementu danych A, jeżeli dla każdej wartości A istnieje jedna jednoznacznie określona wartość B.

Element danych B jest niefunkcyjnie zależny od elementy danych A, jeżeli dla każdej wartości elementu danych A istnieje ograniczony zbiór wartości elementu B.

- 1 PN, 2 PN, 3 PN i BCNF dotyczą zależności funkcyjnych
- 4 PN oraz 5 PN dotyczą zależności niefunkcyjnych
6.3.1. Zależności funkcyjne

Najważniejszy rodzaj więzów, z jakimi mamy do czynienia w modelu relacyjnym, dotyczy więzów jednoznaczności, które nazywa się również zależnością funkcyjną.

Nie istnieją metody pozwalające automatycznie określić zależności funkcyjne, aby to uczynić należy dokładnie przeanalizować znaczenie wszystkich atrybutów.

### Filmy

<table>
<thead>
<tr>
<th>Tytuł</th>
<th>Rok</th>
<th>Długość</th>
<th>Typ</th>
<th>Producent</th>
<th>Gwiazda</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gwiezdne Wojny</td>
<td>1977</td>
<td>124</td>
<td>kolor</td>
<td>Fox</td>
<td>Carrie Fisher</td>
</tr>
<tr>
<td>Gwiezdne Wojny</td>
<td>1977</td>
<td>124</td>
<td>kolor</td>
<td>Fox</td>
<td>Mark Hamil</td>
</tr>
<tr>
<td>Gwiezdne Wojny</td>
<td>1977</td>
<td>124</td>
<td>kolor</td>
<td>Fox</td>
<td>Harrison Ford</td>
</tr>
<tr>
<td>Potężne Kaczory</td>
<td>1991</td>
<td>104</td>
<td>kolor</td>
<td>Disney</td>
<td>Emilio Estavez</td>
</tr>
</tbody>
</table>

W relacji można wyodrębnić zależności:

- tytuł, rok → długość
- tytuł, rok → typ
- tytuł, rok → producent

Ponieważ wszystkie zależności maja po lewej stronie te same atrybuty można je zapisać:

- tytuł, rok → długość, typ, producent

Zależności funkcyjne (podobnie jak inne więzy) dotyczą schematu bazy danych, a nie określonej instancji
Reguły dotyczące zależności funkcyjnych

🌟 Reguła przechodniości
Aksjomaty Armstronga

✿ Zbiór atrybutów określa funkcyjnie dowolny jego podzbiór  
  • np.: nr, nazwisko → nazwisko  

✿ Jeżeli zbiór atrybutów X funkcyjnie określa zbiór atrybutów Y oraz Z jest innym 
  zbiorem atrybutów wówczas suma zborów X i Z funkcyjnie określa sumę Y i Z  
  X → Y ⇒ X∪Z → Y∪Z  
  • np.: nr → nazwisko to zachodzi również nr, data → nazwisko, data 

✿ Jeżeli  X→Y i Y→Z ⇒ X→Z  
  ▪ X→Y ⇒ X∪Z → Z  
  ▪ X→Y i X→Z ⇔ X → Y∪Z  
  ▪ Jeżeli A={A₁, A₂ ... Aₙ} to X→A ⇔ X→A₁ ... X→Aₙ  

(Zwrotność)  
(Rozszerzenie)  
(Przechodniość)
Obliczanie domknięcia zbioru atrybutów

**Bardzo często należy określić, które pojedyncze atrybuty są funkcyjnie zależne od danego zbioru atrybutów.**

**Algorytm X-domknięcia:**

Krok 1: \( X(0) = X, n=0 \)

Krok 2: Jeżeli istnieje zależność \( A \rightarrow B \) oraz \( A \subseteq X(n) \) i \( B \not\subseteq X(n) \) to \( X(n+1) = X(n) \cup B \).

W przeciwnym wypadku zakończ algorytm

Krok 3: \( n=n+1 \) i wróć do kroku 2.

Rozważmy relację

Należności(nazwisko, ulica, miasto, województwo, data, wielkość)

Z następującymi zależnościami funkcyjnymi:

nazwisko → ulica, miasto, województwo
nazwisko, data → wielkość
miasto → województwo

Wykonując algorytm X-domknięcia otrzymamy:

\[
X(0) = \{\text{nazwisko}, \text{data}\} \\
X(1) = \{\text{nazwisko}, \text{ulica}, \text{miasto}, \text{województwo}, \text{data}\} \\
X(2) = \{\text{nazwisko}, \text{ulica}, \text{miasto}, \text{województwo}, \text{data}, \text{wielkość}\} \\
X(3) = X(2)
\]
6.4. Pierwsza postać normalna 1PN

Relacja (tablica) jest w pierwszej postaci normalnej (1PN) wtedy i tylko wtedy, gdy każdy atrybut niekluczowy jest funkcjonalnie zależny od klucza głównego

- Pierwsza postać normalna to warunek, że wszystkie wartości kolumn muszą być elementarne
- Elementarne znaczy w tym przypadku niepodzielne. 1 PN wymaga, żeby dla każdej pozycji wiersz-kolumna w tablicy istniała tylko jedna wartość, a nie tablica lub lista wartości.
- Jeśli w kolumnie przechowuje się całe listy wartości, wtedy trudno jest nimi operować.
- 1 PN zabrania także istnienia powtarzających się grup, nawet jeśli miałyby być one złożone z kolumn elementarnych
Przejście do 1 PN – przykład

<table>
<thead>
<tr>
<th>Moduły</th>
<th>KP</th>
<th>NazwaModułu</th>
<th>NrPrac</th>
<th>Pracownik</th>
<th>NrStud</th>
<th>Student</th>
<th>Ocena</th>
<th>TypOceny</th>
</tr>
</thead>
<tbody>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>244</td>
<td>Buczek Jan</td>
<td>34698</td>
<td>Kowalski H.</td>
<td>4.0</td>
<td>zal.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Projektowanie relacyjnych baz danych</td>
<td>244</td>
<td>Buczek Jan</td>
<td>34698</td>
<td>Kowalski H.</td>
<td>3.0</td>
<td>zal.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Obiektowe bazy danych</td>
<td>445</td>
<td>Kalita Henryk</td>
<td>35785</td>
<td>Woś S.</td>
<td>3.5</td>
<td>egz.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Atrybuty NrStud, Student, Ocena oraz TypOceny nie są funkcjonalnie zależne od klucza głównego, pozostałe są zależne. Należy więc utworzyć dwie tabele:
- jedną dla atrybutów zależnych
- drugą dla funkcjonalnie niezależnych atrybutów

<table>
<thead>
<tr>
<th>Moduły</th>
<th>KP</th>
<th>NazwaModułu</th>
<th>NrPrac</th>
<th>Pracownik</th>
<th>NrStud</th>
<th>Student</th>
<th>Ocena</th>
<th>TypOceny</th>
</tr>
</thead>
<tbody>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>244</td>
<td>Buczek Jan</td>
<td>34698</td>
<td>Kowalski H.</td>
<td>4.0</td>
<td>zal.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>34610</td>
<td>Lech M.</td>
<td>34698</td>
<td>Kowalski H.</td>
<td>5.0</td>
<td>zal.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>34698</td>
<td>Buczek Jan</td>
<td>34610</td>
<td>Lech M.</td>
<td>3.5</td>
<td>egz.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Projektowanie relacyjnych baz danych</td>
<td>34698</td>
<td>Buczek Jan</td>
<td>34698</td>
<td>Kowalski H.</td>
<td>3.0</td>
<td>zal.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Projektowanie relacyjnych baz danych</td>
<td>34698</td>
<td>Buczek Jan</td>
<td>34698</td>
<td>Kowalski H.</td>
<td>4.0</td>
<td>egz.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Obiektowe bazy danych</td>
<td>35785</td>
<td>Kalita Henryk</td>
<td>34610</td>
<td>Lech M.</td>
<td>4.0</td>
<td>egz.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Atrybuty NazwaModułu, NrStud i TypOceny utworzą klucz główny tabeli Oceny
6.5. Druga postać normalna 2PN

Relacja jest w drugiej postaci normalnej wtedy i tylko wtedy, gdy jest w 1PN i każdy atrybut niekluczowy jest w pełni funkcyjnie zależny od klucza głównego.

❖ Tablica ma drugą postać normalną, jeśli jest w 1 PN i każda kolumna nie należąca do żadnego klucza potencjalnego jest całkowicie zależna od klucza głównego.

❖ Innymi słowy, tablice powinny przechowywać dane dotyczące tylko jednej „rzeczy” (jednostki, obiektu, zdarzenia) oraz ta „rzecz” powinna być opisywalna przez jej klucz główny.

❖ Tablica która jest w 1PN może nie być w 2PN tylko wtedy gdy posiada klucz główny złożony bo tylko wówczas któryś z atrybutów może być identyfikowany przez część klucza.
Przejście do 2 PN – przykład

### Oceny

<table>
<thead>
<tr>
<th>KP</th>
<th>NazwaModułu</th>
<th>NrStud</th>
<th>TypOceny</th>
<th>Student</th>
<th>Ocena</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Systemy relacyjnych baz danych</td>
<td>34698</td>
<td>zal.</td>
<td>Kowalski H.</td>
<td>4.0</td>
</tr>
<tr>
<td></td>
<td>Systemy relacyjnych baz danych</td>
<td>34698</td>
<td>egz.</td>
<td>Kowalski H.</td>
<td>3.5</td>
</tr>
<tr>
<td></td>
<td>Systemy relacyjnych baz danych</td>
<td>34610</td>
<td>zal.</td>
<td>Lech M.</td>
<td>5.0</td>
</tr>
<tr>
<td></td>
<td>Projektowanie relacyjnych baz danych</td>
<td>34698</td>
<td>zal.</td>
<td>Kowalski H.</td>
<td>3.0</td>
</tr>
<tr>
<td></td>
<td>Projektowanie relacyjnych baz danych</td>
<td>34698</td>
<td>egz.</td>
<td>Kowalski H.</td>
<td>4.0</td>
</tr>
<tr>
<td></td>
<td>Obiektowe bazy danych</td>
<td>35785</td>
<td>egz.</td>
<td>Woś S.</td>
<td>3.5</td>
</tr>
</tbody>
</table>

Tabela **Oceny** (niebieska) nie jest w 2PN gdyż pole **Student** nie zależy od całego klucza głównego a jedynie od atrybutu **NrStud**

### Studenci

<table>
<thead>
<tr>
<th>NrStud</th>
<th>Student</th>
</tr>
</thead>
<tbody>
<tr>
<td>34698</td>
<td>Kowalski H.</td>
</tr>
<tr>
<td>34610</td>
<td>Lech M.</td>
</tr>
<tr>
<td>35785</td>
<td>Woś S.</td>
</tr>
</tbody>
</table>

### Moduły

<table>
<thead>
<tr>
<th>KP</th>
<th>NazwaModułu</th>
<th>NrPrac</th>
<th>Pracownik</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Systemy relacyjnych baz danych</td>
<td>244</td>
<td>Buczek Jan</td>
</tr>
<tr>
<td></td>
<td>Projektowanie relacyjnych baz danych</td>
<td>244</td>
<td>Buczek Jan</td>
</tr>
<tr>
<td></td>
<td>Obiektowe bazy danych</td>
<td>445</td>
<td>Kalita Henryk</td>
</tr>
</tbody>
</table>

Bazy danych Wykład 3
6.6. Trzecia postać normalna 3PN

Relacja jest w trzeciej postaci normalnej wtedy i tylko wtedy, gdy jest w 2PN i każdy atrybut niekluczowy jest bezpośrednio zależny od klucza głównego.

Tablica jest w trzeciej postaci normalnej jeśli jest w 2 PN i wszystkie kolumny nie należące do żadnego klucza potencjalnego są wzajemnie niezależne.
Przejście do 3 PN – przykład

<table>
<thead>
<tr>
<th>Moduły</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NazwaModułu</td>
<td>NrPrac</td>
<td>Pracownik</td>
</tr>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>244</td>
<td>Buczek Jan</td>
</tr>
<tr>
<td>Projektowanie relacyjnych baz danych</td>
<td>244</td>
<td>Buczek Jan</td>
</tr>
<tr>
<td>Obiektowe bazy danych</td>
<td>445</td>
<td>Kalita Henryk</td>
</tr>
</tbody>
</table>

Jedynie tabela Moduły nie jest w 3PN gdyż pole Pracownik zależy od pola NrPrac a nie bezpośrednio od pola NazwaModułu.

Należy dokonać podziału tabeli

<table>
<thead>
<tr>
<th>Tabele w 3 PN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moduły</td>
</tr>
<tr>
<td>--------</td>
</tr>
<tr>
<td>NazwaModułu</td>
</tr>
<tr>
<td>Systemy relacyjnych baz danych</td>
</tr>
<tr>
<td>Projektowanie relacyjnych baz danych</td>
</tr>
<tr>
<td>Obiektowe bazy danych</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Moduły</th>
<th>Oceny</th>
<th>Studenci</th>
</tr>
</thead>
<tbody>
<tr>
<td>NazwaModułu</td>
<td>NrStud</td>
<td>TypOceny</td>
</tr>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>34698</td>
<td>zal.</td>
</tr>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>34698</td>
<td>egz.</td>
</tr>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>34610</td>
<td>zal.</td>
</tr>
<tr>
<td>Projektowanie relacyjnych baz danych</td>
<td>34698</td>
<td>zal.</td>
</tr>
<tr>
<td>Projektowanie relacyjnych baz danych</td>
<td>34698</td>
<td>egz.</td>
</tr>
<tr>
<td>Obiektowe bazy danych</td>
<td>35785</td>
<td>egz.</td>
</tr>
</tbody>
</table>
**Metoda Bersteina**

Dowolną relację można przekształcić do 3PN korzystając z metody Bersteina:

**Krok 1.** Przekształć każdą zależność tak aby po prawej stronie był tylko jeden atrybut.

**Krok 2.** Wyeliminuj powtarzające się zależności

**Krok 3.** Przekształć zależność tak aby żaden podzbiór atrybutów stojący po lewej stronie nie określił prawej strony zależności.

**Krok 4.** Połącz zależności z takimi samymi lewymi stronami

**Krok 5.** Znajdź klucz dla relacji pierwotnej. Jeżeli żaden z kluczy nie jest zawarty w zbiorach z kroku poprzedniego to utwórz nowy zbiór z atrybutami klucza.

**Krok 6.** Jeżeli jakiś nowo utworzony zbiór jest rzutem (projekcją) innego, to wyeliminuj ten zbiór

**Rozważmy relację**

Magazyn(nr, mistrz, wydział, materiał, ilość, cena)

Z następującymi zależnościami funkcyjnymi:

nr → mistrz
mistrz → wydział
mistrz, materiał → ilość
materiał → cena

**Krok 1.** Zależności funkcyjne spełniają wymagania

**Krok 2.** Brak zależności redundancyjnych

**Krok 3.** Lewe strony są minimalne. Żaden podzbiór atrybutów z lewej strony nie określa prawej strony

**Krok 4,5.** Otrzymujemy pięć relacji

R1(nr, mistrz)
R2(mistrz, wydział)
R3(mistrz, materiał, ilość)
R4(materiał, cena)
R5(nr, materiał)

**Krok 6.** Żadna z otrzymanych relacji nie jest rzutem innej *nie zawiera się w niej*
6.7. Diagramy zależności

*Klasyczna normalizacja* opisana jako proces rozkładu odwracalnego ma kilka wad:

- wymaga aby zbiór danych był w pełni określony,
- jest bardzo czasochłonna.

*Alternatywą dla klasycznej normalizacji mogą być diagramy zależności.*

*Zalety diagramów zależności:*

- nie wymagają pełnego określenia danych
- czytelny zapis graficzny
6.7.2. Pragmatyka rysowania diagramów zależności

Zależność między dwoma elementami danych można rysować tylko w jedną stronę (od A do B albo od B do A).

Jeżeli między elementami istnieje w jedną stronę zależność funkcyjna a w drugą niefunkcyjna to wybieramy kierunek zależności funkcyjnej.
  - Np.: w kierunku od NrPrac do NazwaWydziału jest zależność funkcyjna a w kierunku NazwaWydziału do NrPrac niefunkcyjna.

Jeżeli w obu kierunkach występują zależności tego samego typu to wybieramy kierunek, który dla nas jest wygodniejszy.

Zależności mogą być złożone – gdy złożenie kilku elementów determinuje jakiś inny element

Jeżeli A determinuje B a b determinuje C to mamy doczynienia z zależnością przechodnią. Wykrycie i usunięcie tego typu zależności jest ważnym elementem procesu normalizacji.
6.8 Akomodacja.

Akomodacja to proces przekształcania diagramu zależności w schemat relacyjny

Reguła Boyce’a-Codd

Każdy funkcjonalnie determinujący element staje się kluczem kandydującym tabeli. Wszystkie bezpośrednio zależne od niego elementy danych stają się niegłównymi atrybutami tabeli

Tabele:
Pracownicy(NrPrac, Pracownik, Dział)
Działy(Dział, Lokalizacja)

Elementy NrPrac oraz Pesel są kluczami kandydującymi

* Liczba elementów determinujących (z których wychodzi strzałki) wskazuje liczbę wymaganych tabel
* Element do którego wchodzi strzałka i z którego wychodzi strzałka stanowi klucz obcy
Każdy niefunkcyjnie determinujący element staje się częścią klucza głównego tabeli.

Dokładnie tworzymy klucz główny z determinującego elementu danych i zależnych elementów danych wchodzących w skład związku niefunkcyjnego.

Otrzymujemy tabele:
- Pracownicy(NrPrac, Pracownik, Dział ...)
- Działy(Dział, Lokalizacja ...)
- Język(NrPrac, JęzykPrac ...)
6.9. Rysowanie diagramów zależności i postacie normalne

**Moduły**

<table>
<thead>
<tr>
<th>KP</th>
<th>NazwaModułu</th>
<th>NrPrac</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Systemy relacyjnych baz danych</td>
<td>244</td>
</tr>
<tr>
<td></td>
<td>Projektowanie relacyjnych baz danych</td>
<td>244</td>
</tr>
<tr>
<td></td>
<td>Obiektowe bazy danych</td>
<td>445</td>
</tr>
</tbody>
</table>

**Wykładowcy**

<table>
<thead>
<tr>
<th>NrPrac</th>
<th>Pracownik</th>
</tr>
</thead>
<tbody>
<tr>
<td>244</td>
<td>Buczek Jan</td>
</tr>
<tr>
<td>445</td>
<td>Kalita Henryk</td>
</tr>
</tbody>
</table>

**Studenci**

<table>
<thead>
<tr>
<th>NrStud</th>
<th>Student</th>
</tr>
</thead>
<tbody>
<tr>
<td>34698</td>
<td>Kowalski H.</td>
</tr>
<tr>
<td>34610</td>
<td>Lech M.</td>
</tr>
<tr>
<td>35785</td>
<td>Woś S.</td>
</tr>
</tbody>
</table>

**Oceny**

<table>
<thead>
<tr>
<th>NazwaModułu</th>
<th>NrStud</th>
<th>TypOceny</th>
<th>Ocena</th>
</tr>
</thead>
<tbody>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>34698</td>
<td>zal.</td>
<td>4.0</td>
</tr>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>34698</td>
<td>egz.</td>
<td>3.5</td>
</tr>
<tr>
<td>Systemy relacyjnych baz danych</td>
<td>34610</td>
<td>zal.</td>
<td>5.0</td>
</tr>
<tr>
<td>Projektowanie relacyjnych baz danych</td>
<td>34698</td>
<td>zal.</td>
<td>3.0</td>
</tr>
<tr>
<td>Projektowanie relacyjnych baz danych</td>
<td>34698</td>
<td>egz.</td>
<td>4.0</td>
</tr>
<tr>
<td>Obiektowe bazy danych</td>
<td>35785</td>
<td>egz.</td>
<td>3.5</td>
</tr>
</tbody>
</table>

**Bazy danych Wykład 3**
6.10. Postać normalna Boyce’a-Codda (BCNF)

Relacja jest w postaci normalnej Boyce’a-Codda, jeśli tylko z nietrywialnych zależności wynika, że pewien nadklucz wyznacza funkcyjnie jakiś inny atrybut.

🌟 Zależność funkcyjna nietrywialna – zależność funkcyjna w której co najmniej jeden z atrybutów typu B znajduje się pośród atrybutów typu A
- Np.: tytuł, rok → rok, długość

🌟 Nadklucz – zbiór atrybutów który zawiera klucz.
Przykład

🌟 Założmy, iż mamy zamodelować sytuację:
- każdy student może specjalizować się w kilku dziedzinach,
- student ma jednego asystenta w każdej dziedzinie,
- każda dziedzina ma kilku asystentów, ale jeden asystent doradza tylko w jednej dziedzinie,
- każdy asystent doradza kilu studentom w jednej dziedzinie.

🌟 Przedstawiony schemat spełnia wymogi 3PN
Przykład c.d.

Zauważmy że pomimo normalizacji do 3PN występują anomalie:

- zmiana specjalizacji przez studenta 123456 powoduje utratę informacji o NrPrac 234,
- nie można wstawić informacji o NrPrac 789, który jest asystentem z Informatyki tak długo jak długo jakiś student nie wybierze tej specjalności,
- usunięcie studenta 345678 powoduje jednoczesne usunięcie informacji o Nr 456.

Należy dokonać dekompozycji tabeli Specjalizacje:

- Schemat 1
  AsystenciStudenta(NrStudenta, NrPrac)
  DziedzinyAsystenta(NrPrac, Dziedzina)

- Schemat 2
  AsystenciStudenta(NrStudenta, Dziedzina)
  DziedzinyAsystenta(NrPrac, Dziedzina)
6.11. Czwarta postać normalna (4PN)

Relacja jest w czwartej postaci normalnej, wtedy i tylko wtedy gdy z nietrywialnych zależności wielowartościowych wynika, że pewien nadklucz wyznacza jakiś inny atrybut.

🌟 Relacja jest w 4PN, gdy zbiór atrybutów \( X \) określa wielowartościowo (niefunkcyjnie) \( Y \) to zachodzi jeden z następujących warunków:

- \( Y \) jest puste lub zawiera się w \( X \),
- Suma zbiorów \( X \) i \( Y \) jest kompletnym zbiorem atrybutów dla danej relacji,
- \( X \) zawiera klucz.

🌟 Aby przejść z 3 PN do 4 PN, szukamy tabel, które zawierają dwie lub więcej niezależnych zależności wielowartościowych.
Przykład

Założmy, iż mamy przechowywać:

- informację o pracownikach,
- umiejętnościach pracowników,
- znajomości językóów przez pracowników.

- Pracownik może posiadać wiele umiejętności oraz znać kilka języków (umiejętności nie są związane ze znajomością języków).

### Diagramy zależności

![Diagramy zależności]

### Tabele przed i po normalizacji do 4PN

#### Pracownicy

<table>
<thead>
<tr>
<th>NrPrac</th>
<th>Umiejętność</th>
<th>Język</th>
</tr>
</thead>
<tbody>
<tr>
<td>123456</td>
<td>Obsługa komputera</td>
<td>angielski</td>
</tr>
<tr>
<td>123456</td>
<td>Obsługa komputera</td>
<td>francuski</td>
</tr>
<tr>
<td>123456</td>
<td>Prawo jazdy</td>
<td>angielski</td>
</tr>
<tr>
<td>234567</td>
<td>Prawo jazdy</td>
<td>niemiecki</td>
</tr>
<tr>
<td>234567</td>
<td>Obsługa komputera</td>
<td>angielski</td>
</tr>
</tbody>
</table>

#### Umiejętności

<table>
<thead>
<tr>
<th>NrPrac</th>
<th>Umiejętność</th>
</tr>
</thead>
<tbody>
<tr>
<td>123456</td>
<td>Obsługa komputera</td>
</tr>
<tr>
<td>123456</td>
<td>Prawo jazdy</td>
</tr>
<tr>
<td>234567</td>
<td>Prawo jazdy</td>
</tr>
<tr>
<td>234567</td>
<td>Obsługa komputera</td>
</tr>
</tbody>
</table>

#### Języki

<table>
<thead>
<tr>
<th>NrPrac</th>
<th>Język</th>
</tr>
</thead>
<tbody>
<tr>
<td>123456</td>
<td>angielski</td>
</tr>
<tr>
<td>123456</td>
<td>francuski</td>
</tr>
<tr>
<td>234567</td>
<td>niemiecki</td>
</tr>
<tr>
<td>234567</td>
<td>angielski</td>
</tr>
</tbody>
</table>
6.12. Piątą postać normalną (5PN)

Relacja jest w piątej postaci normalnej jeżeli jest w 4 PN i nie istnieje jej rozkład odwracalny na zbiór mniejszych tabel

Założmy, iż mamy przechowywać:
- informację o dealerach samochodów,
- produktach sprzedawanych przez dealerów,

<table>
<thead>
<tr>
<th>PunktySprzedarzy</th>
<th>Dealer</th>
<th>Producent</th>
<th>Typ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto Zbyt</td>
<td>Ford</td>
<td>osobowy</td>
<td></td>
</tr>
<tr>
<td>Auto Zbyt</td>
<td>Opel</td>
<td>dostawczy</td>
<td></td>
</tr>
<tr>
<td>Cztery Kółka</td>
<td>Ford</td>
<td>dostawczy</td>
<td></td>
</tr>
<tr>
<td>Cztery Kółka</td>
<td>Opel</td>
<td>osobowy</td>
<td></td>
</tr>
</tbody>
</table>

Dealerzy reprezentują firmy, firmy wytwarzają produkty i dealerzy sprzedają te produkty (diagram zależności). Nie można dokonać rozkładu struktury pokazanej na diagramie gdyż np.: Auto Zbyt sprzedaje samochody osobowe marki Ford i dostawcze marki Opel, a nie sprzedaje dostawczych fordów ani osobowych opli.