

## **Zadanie**

**Zaprojektuj dowolną bazę danych dokumentując jej powstawanie wg wzoru, gotowa baza i plik .doc dokumentujący jej powstawanie podlegają sprawdzeniu i ocenie.**

## **Przykład**

### **1. Wprowadzenie**

#### **1.1 Podstawowe pojęcia dotyczące relacyjnych baz danych**

Baza danych jest uporządkowanym zbiorem danych o określonej strukturze, przechowywanym na nośniku informacji. System baz danych można zdefiniować jako bazę danych wraz z oprogramowaniem umożliwiającym operowanie tymi danymi. Zadaniem bazy danych jest przechowywanie informacji o pewnych typach obiektów. W języku baz danych obiekty te są nazywane encjami.

Wśród baz danych można wyróżnić relacyjne i jednorodny bazy danych. Relacja to związek (powiązanie) pomiędzy różnymi tabelami w bazie za pomocą połączonych pól. Relacyjna baza danych składa się z wielu tabel powiązanych ze sobą relacjami, zdefiniowanymi przez jej twórcę. Pozwala to uniknąć wystąpienia nadmiarowych danych i rozmaitych anomalii które pojawiają się w jednorodnych bazach tzn. zbudowanych z jednej tabeli.

#### **1.2 Zadania postawione przed naszą bazą danych**

Głównym zadaniem jest umożliwienie gromadzenia danych o wynikach studentów. System ma pozwolić na wprowadzanie nowych danych dotyczących wyników osiągniętych przez studenta danego roku studiów, danej grupy w ramach danego przedmiotu podczas ćwiczeń, laboratoriów, zaliczeń i egzaminów w rejestrowanym czasie oraz łatwe uzupełnianie, zmianę i usuwanie tych danych.

Założenia:

- student jest identyfikowany przez unikalny 6-cio znakowy kod (numer albumu);
- jeden student może studiować na wielu kierunkach przy czym jego numer albumu pozostanie taki sam;
- dany moduł może być prowadzony tylko przez jednego wykładowcę;
- tablica MODUŁY przechowuje dane dotyczące wykładów jak i ćwiczeń oraz laboratoriów.

## 2. Schemat bazy danych STUDENT

### 2.1 Jednorodna baza danych

Jednorodna baza danych STUDENT jest bazą składającą się z tylko jednej tabeli zawierającej wszystkie atrybuty dotyczące danych jakie chcemy w niej przechowywać. Baza ta zawiera wiele nie powiązanych ze sobą w sposób bezpośredni kolumn. Przykładowy wygląd tej bazy przedstawiamy poniżej:

#### **STUDENT**

Nazwa\_przedmiotu  
Imię\_wykładowcy  
Nazwisko\_wykładowcy  
Stopień\_naukowy  
Temat\_ćwiczeń  
Nr\_albumu  
Typ\_studiów  
Rok\_studiów  
Nr\_grupy  
Imię\_studenta  
Nazwisko\_studenta  
Typ\_zajęć  
Adres\_email\_studenta  
Ocena\_zaliczająca\_ćwiczenie  
Ocena\_cząstkowa  
Ocena\_końcowa  
Data\_oceny\_ćwiczeń  
Data\_wystawienia\_oceny\_końcowej  
Data\_ćwiczeń  
Typ\_oceny  
Adres\_email\_wykładowcy  
Nr\_ćwiczenia  
Nr\_wykładowcy

Kluczem głównym w tej bazie ustanawiamy Nr\_albumu studenta, dzięki czemu uzyskamy później rozkład podanej tablicy. Po analizie powyższej tabeli można wywnioskować że nie będzie ona efektywnie spełniała założeń bazy. Bez trudu można dostrzec szereg anomalii wynikających z takiego sposobu jej zaprojektowania. Najbardziej widoczną wadą będą powtarzające się dane. Będzie tak przykładowo w przypadku informacji o wykładowcach. Dany wykładowca może prowadzić wiele różnego typu zajęć, co będzie się wiązać z koniecznością wielokrotnego powtarzania jego danych personalnych. W tym miejscu widzimy już o ile bardziej efektywne byłoby stworzenie osobnej tabeli zawierającej jedynie informacje dotyczące prowadzących zajęcia.

Kolejnym problemem pojawiającym się przy jednorodnej bazie STUDENT jest tzw. anomalia przy usuwaniu danych dodatkowo mogąca wymuszać konieczność aktualizacji. Przykładowo: jeśli dany wykładowca z jakichś powodów nie może prowadzić zajęć i wprowadzamy jego zastępcę to z naszej bazy danych musimy całkowicie usunąć dane dotyczące pierwszego z nich pole po polu, a następnie tą samą metodą uzupełnić je nowymi danymi. W przypadku gdy posiadalibyśmy osobną tabelę zawierającą wyłącznie dane wykładowców operacja ta polegałaby jedynie na zmianie pewnego numeru przypisanego konkretnemu wykładowcy.

## 2.2 Normalizacja bazy danych STUDENT

Normalizacja, czyli proces sprowadzania bazy do odpowiedniej postaci polega przede wszystkim na dzieleniu jednorodnej tabeli na kilka połączonych kluczem mniejszych tabel. Głównym powodem, dla którego normalizuje się bazę jest występowanie problemów (opisanych częściowo przez nas w poprzednim punkcie) w przypadku źle zaprojektowanej struktury. Wyróżniamy pięć poziomów normalizacji z czego my zajmiemy się trzema pierwszymi.

### PIERWSZA POSTAĆ NORMALNA

W każdej komórce tabeli znajduje się tylko jedna wartość, czyli: każdy atrybut niekluczowy (nie należący do żadnego klucza) jest funkcyjnie zależny od klucza głównego.

Naszą jednorodną bazę danych podzieliliśmy na dwie tabele obierając jako klucz główny pole o nazwie Nazwa\_przedmiotu:

#### PRZEDMIOTY

Nazwa_przedmiotu
Temat_ćwiczenia
Typ_zajęć
Nr_wykładowcy
Imię_wykładowcy
Nazwisko_wykładowcy
Stopień_naukowy
Email_wykładowcy

#### ZALICZENIA

Nazwa_przedmiotu
Nr_albumu
Imię_studenta
Nazwisko_studenta
Email_studenta
Rok_studiów
Typ_studiów
Nr_grupy
Ocena_cząstkowa
Ocena_końcowa
Ocena_zaliczająca_ćwiczenia
Data_ćwiczeń
Data_oceny_ćwiczeń
Nr_ćwiczenia
Data_wystawienia_zaliczenia

Kolorem czerwonym zazaczyliśmy klucze podstawowe.

Tak wyglądają tabele po zastosowaniu pierwszego kroku normalizacji. Kolejnym etapem będzie normalizacja do poziomu drugiego.

### DRUGA POSTAĆ NORMALNA

Aby przejść do drugiej postaci normalnej baza musi być w pierwszej postaci normalnej oraz kolumna nie należąca do klucza nie może być zależna od części klucza głównego, czyli: każdy atrybut niekluczowy jest w pełni zależny funkcyjnie od całości klucza głównego.

W tabeli Zaliczenia klucz główny składał się z trzech atrybutów, przy czym każdy atrybut nie będący kluczem był bezpośrednio zależny tylko od jednej z części klucza. Dlatego sprowadzając bazę do drugiej postaci normalnej, podzieliliśmy tę tabelę na trzy osobne, w których klucz główny był jedną z tych właśnie części klucza w tabeli bazowej:

#### PRZEDMIOTY

Nazwa_przedmiotu
Nr_wykładowcy
Nazwisko_wykładowcy
Imię_wykładowcy
Email_wykładowcy
Typ_zajęć
Stopień_naukowy

#### STUDENT

Nr_albumu
Imię_studenta
Nazwisko_studenta
Email_studenta
Rok_studiów
Typ_studiów
Nr_grupy

#### ZALICZENIE\_MODULU

Nazwa_przedmiotu
Nr_albumu
Data_wystawienia_oceny_końcowej
Ocena_końcowa

#### ZALICZENIE\_ĆWICZEŃ

Nr_ćwiczenia
Nazwa_przedmiotu
Nr_albumu
Temat_ćwiczenia
Data_ćwiczeń
Ocena_cząstkowa
Ocena_zaliczająca
Data_oceny_cząstkowej
Data_wystawienia_zaliczenia

Kolorem zielonym są oznaczone klucze obce.

W ten sposób otrzymaliśmy powyższe cztery tabele, będące reprezentacją drugiej postaci normalnej.

#### TRZECIA POSTAĆ NORMALNA

Aby przejść do trzeciej postaci normalnej baza musi być w drugiej postaci normalnej oraz każda kolumna nie należąca do klucza nie może być zależna od innej kolumny nie należącej do klucza, czyli: każdy niekluczowy atrybut musi być bezpośrednio zależny od klucza głównego.

Nasza baza nie spełnia warunków trzeciej postaci normalnej ponieważ w tabeli PRZEDMIOTY są pola nie zależne bezpośrednio od klucza głównego - Nazwa\_przedmiotu. Jest tak w przypadku atrybutów związanych z Wykładowcą, czyli: Nr\_wykładowcy, Nazwisko\_wykładowcy, Imię\_wykładowcy, Email\_wykładowcy, Stopień\_naukowy. Dlatego,

sprowadzając naszą bazę do trzeciej postaci normalnej wprowadziliśmy piątą tabelę przechowującą informacje dotyczące danych personalnych każdego z wykładowców.

Ostatecznie po procesie normalizacji nasza baza danych składa się z pięciu tabel o nazwach: STUDENT, WYKŁADOWCA, MODUŁY, ZALICZENIE\_MODUŁU, ZALICZENIE\_ĆWICZENIA.

#### WYKŁADOWCY

ID_wykładowcy
Nazwisko_wykładowcy
Imię_wykładowcy
Stopień_naukowy
Email_wykładowcy

#### STUDENT

Nr_albumu
Imię_studenta
Nazwisko_studenta
Email_studenta
Rok_studiów
Typ_studiów
Nr_grupy

#### ZALICZENIE\_MODUŁU

ID_zaliczenia
Nr_albumu
ID_modułu
Ocena_końcowa
Data_wystawienia_oceny_końcowej

#### ZALICZENIE\_ĆWICZEŃ

Nr_ćwiczenia
ID_modułu
Nr_albumu
Temat_ćwiczenia
Data_ćwiczeń
Ocena_częstkowa
Ocena_zaliczająca
Data_oceny_częstkowej
Data_wystawienia_zaliczenia

#### MODUŁY

ID_modułu
ID_wykładowcy
Nazwa_przedmiotu
Typ_zajęć

Nazwę tabeli PRZEDMIOTY zamieniliśmy na MODUŁY, a nazwę atrybutu Nr\_wykładowcy na ID\_wykładowcy.

Wprowadziliśmy również nowe pole o nazwie ID\_modułu jako klucz główny w tabeli MODUŁY ponieważ przyjęte przez nas założenia sprawiły iż atrybut Nazwa\_przedmiotu nie pozwalałby określić w sposób jednoznaczny każdego wiersza tabeli MODUŁY (jednocześnie stało się ono kluczem obcym w innych tabelach zastępując pole Nazwa\_przedmiotu).

## 2.3 Typy danych i ich rozmiar dla poszczególnych atrybutów klasy encji

Klucze główne zostały zaznaczone kolorem czerwonym, natomiast klucze obce łączące relacje kolorem zielonym.

### a) STUDENT

Nazwa atrybutu	Typ danych	Rozmiar	Uwagi
<b>Nr_albumu</b>	tekst	8	6-cio znakowy identyfikator studenta; może składać się z samych cyfr lub z dwóch liter oraz czterech cyfr
Imię_studenta	tekst	16	zakładamy że imię nie przekracza 16 znaków
Nazwisko_studenta	tekst	40	pole to może zawierać nazwiska dwuczłonowe
Email_studenta	hiperłącze	-	adres typu: <a href="mailto:bbbb@poczta.pl">bbbb@poczta.pl</a>
Rok_studiów	tekst	2	symbol typu: 1d, 2d, 3d, ozn.kolejne lata studiów dziennych oraz 1z, 2z, 3z- zaocznych
Typ_studiów	tekst	15	atrybut ten określa kierunek studiów np. fizyka
Nr_grupy	liczba	1	określa numer grupy dziekanatowej

### b) WYKŁADOWCY

Nazwa atrybutu	Typ danych	Rozmiar	Uwagi
<b>ID_wykładowcy</b>	autonumerowanie	liczba całkowita długa	przypisanie unikalnego numeru wykładowcy
Nazwisko_wykładowcy	tekst	40	pole to może zawierać nazwiska dwuczłonowe
Imię_wykładowcy	tekst	16	zakładamy że imię nie przekracza 16 znaków
Stopień_naukowy	tekst	15	tytuły naukowe jakie posiada dany wykładowca np. prof dr inż. (tylko skróty)
Email_wykładowcy	hiperłącze	-	adres typu: <a href="mailto:bbbb@poczta.pl">bbbb@poczta.pl</a>

### c) MODUŁY

Nazwa atrybutu	Typ danych	Rozmiar	Uwagi
ID_modułu	autonumerowanie	liczba całkowita długa	przypisanie unikalnego numeru przedmiotowi
ID_wykładowcy	liczba	liczba całkowita długa	atrybut ten jako klucz obcy pozwala odwoływać się do tabeli WYKŁADOWCY
Nazwa_przedmiotu	tekst	20	atrybut ten zawiera zarówno nazwy ćwiczeń jak i nazwy wykładów
Typ_zajęć	tekst	12	określa czy jest to: laboratorium, wykład czy ćwiczenie

### d) ZALICZENIE\_MODUŁU

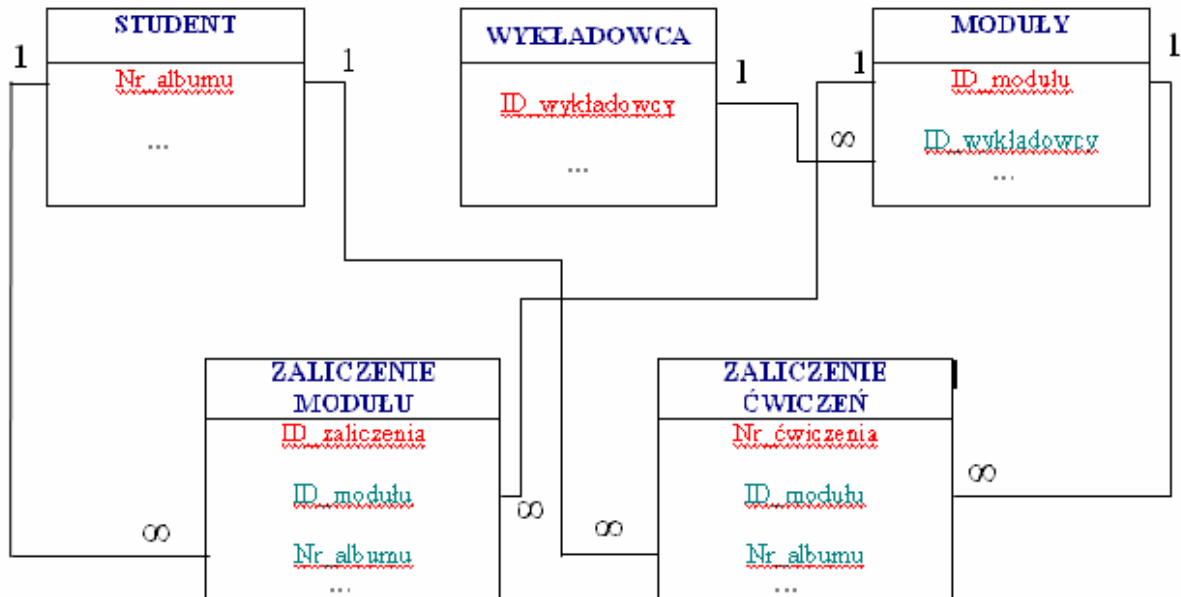
Nazwa atrybutu	Typ danych	Rozmiar	Uwagi
Id_zaliczenia	autonumerowanie	liczba całkowita długa	unikalny numer identyfikujący każde zaliczenie
Nr_albumu	tekst	8	klucz obcy pozwalający na odwołanie się do tablicy STUDENT
ID_modułu	liczba	liczba całkowita długa	klucz obcy wiążący tabele ZALICZENIE_MODUŁU i MODUŁY
Ocena_końcowa	tekst	10	w polu tym znajdować się może ocena uzyskana z egzaminu w skali od 2 do 5 bądź samo słowo zaliczenie w przypadku braku egzaminu
Data_wystawienia oceny_końcowej	data/godź	-	dzień w którym została wystawiona ocena

### e) ZALICZENIE\_ĆWICZEŃ

Nazwa atrybutu	Typ danych	Rozmiar	Uwagi
Nr_ćwiczenia	autonumerowanie	liczba całkowita długa	unikalny numer identyfikujący ćwiczenia dotyczące danego tematu niezależnie od ilości poświęconych mu zajęć
ID_modułu	liczba	liczba całkowita długa	klucz obcy wiążący tabele ZALICZENIE_ĆWICZEŃ i MODUŁY
Nr_albumu	tekst	8	klucz obcy pozwalający na odwołanie się do tablicy STUDENT
Temat_ćwiczenia	tekst	30	temat którego dotyczyły ćwiczenia
Data_ćwiczenia	data/godź	-	data odbycia się ćwiczeń
Ocena_cząstkowa	liczba	2	ilość punktów w skali 1-10
Ocena_zaliczająca	tekst	10	w polu tym znajdować się może ocena uzyskana z egzaminu w skali od 2 do 5 bądź samo słowo zaliczenie
Data_oceny_cząstkowej	data/godź	-	data wystawienia oceny z danego ćwiczenia

## 2.4 Relacje jakie nakładamy na naszą bazę danych

Aby nasza baza spełniała postawione przez nas założenia musimy jeszcze powiązać poszczególne tabele ze sobą, czyli nałożyć odpowiednie relacje. Relacje ułatwią nam przeglądanie, wyszukiwanie potrzebnych nam danych mimo tego że nie będą one znajdowały się w jednej tabeli.



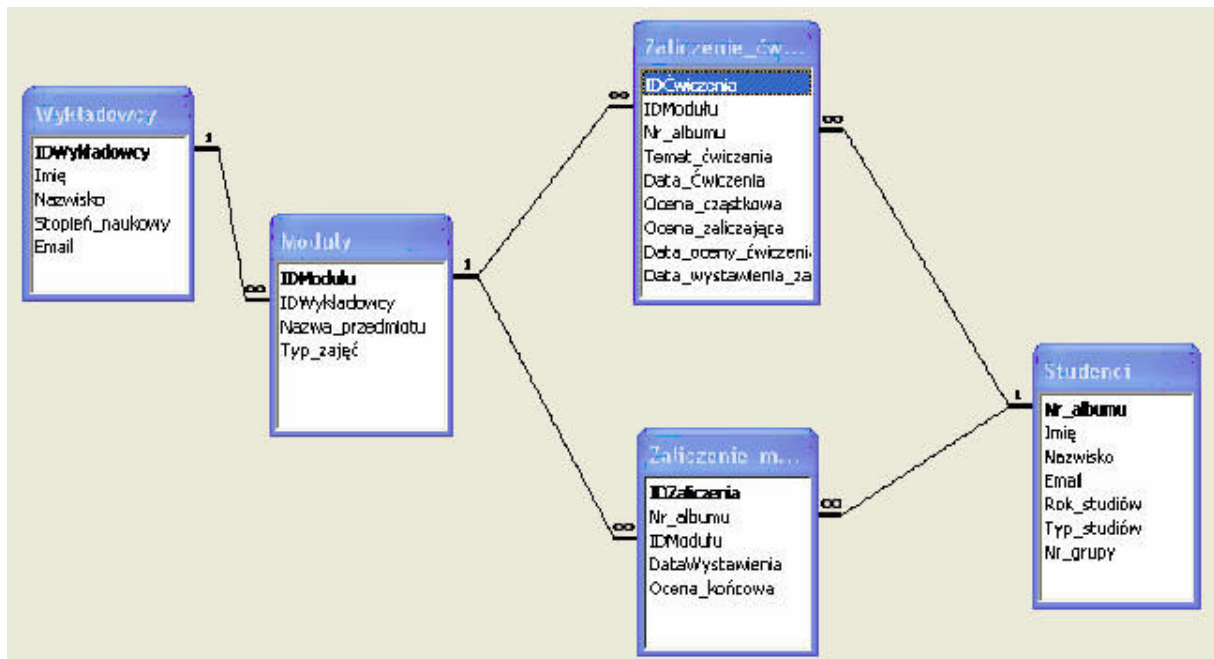
Rys. Schemat relacji jakie nakładamy na bazę.

Relacja pomiędzy tabelami:

- **STUDENT** a **ZALICZENIE\_MODUŁU** - przedmiot jest zaliczany przez konkretnego studenta (identyfikowanego przez Nr\_albumu); jeden student na danym roku studiów zalicza wiele przedmiotów;
- **STUDENT** a **ZALICZENIE\_ĆWICZEŃ** - ćwiczenia są zaliczane przez konkretnego studenta (identyfikowanego przez Nr\_albumu); jeden student na danym roku studiów zalicza ćwiczenia z wielu tematów i wielu przedmiotów. Mówimy tu zarówno o ocenach częściowych jak i końcowych;
- **WYKŁADOWCY** a **MODUŁY** - przedmiot jest prowadzony przez danego wykładowcę o określonych danych personalnych; jeden wykładowca może prowadzić wiele modułów;
- **MODUŁY** a **ZALICZENIE\_MODUŁU**- jeden moduł jest zaliczany przez wielu studentów;
- **MODUŁY** a **ZALICZENIE\_ĆWICZEŃ**- jedno ćwiczenie są zaliczane przez wielu studentów;



Tak wygląda zaproponowany przez nas teoretyczny schemat bazy danych STUDENT. Po przetestowaniu naszego projektu bazy w środowisku Microsoft Access, możemy stwierdzić że jest on poprawny. Oto wygląd tabel powiązanych relacjami zaprojektowany w wymienionym środowisku:



### 3. Formularze

...opisujemy analogicznie jak wyżej

### 4. Kwerendy

...opisujemy analogicznie jak wyżej

### 5. Raporty

...opisujemy analogicznie jak wyżej

Załączona baza danych STUDENT.mdb, wprowadzono przykładowe rekordy do poszczególnych tabel ( Główna - 20, pomocnicze min. 5)