Instytut Sterowania i Systemów Informatycznych
Uniwersytet Zielonogórski

Bazy Danych

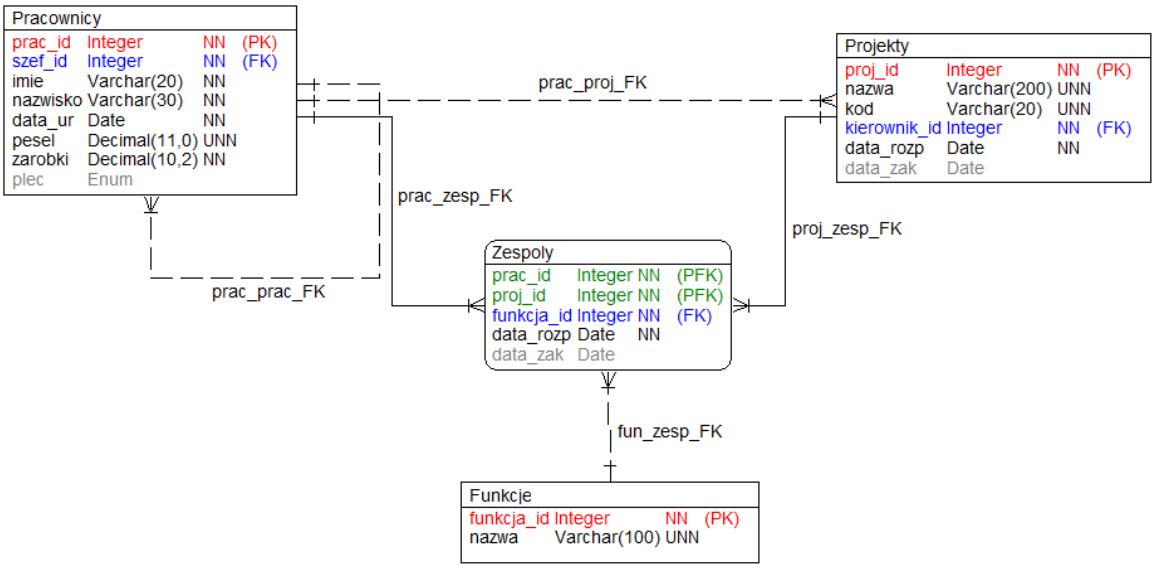
Ćwiczenie 10: Podstawy języka SQL, część 5, polecenia DDL (CREATE, ALTER, DROP)

opracował: dr hab. inż. Artur Gramacki (a.gramacki@issi.uz.zgora.pl)

Uwaga: w ćwiczeniu wszystkie polecenia *SQL-owe* piszemy ręcznie, bez używania narzędzi wspomagających projektowanie, jak na przykład *Toad Data Modeler*!

Pamiętaj, że w swojej (ewentualnej) przyszłej praktyce zawodowej związanej z bazami danych, wielokrotnie będziesz zmuszony wykonać NATYCHMIAST jakąś BARDZO PILNĄ I WAŻNĄ czynność w bazie danych. Z dużym prawdopodobieństwem nie będzie wtedy dostępne żadne inne narzędzie dostępu do bazy poza prostą konsolą tekstową (np. *MySQL Monitor*; program *mysql.exe*). Dlatego niezwykle istotna będzie wtedy umiejętność pisania poleceń SQL w „czystej" postaci bez stosowania dodatkowych narzędzi wspomagających pracę administratora / użytkownika systemu bazodanowego !!!

1. Utworzyć strukturę relacyjną pokazaną na rysunku (oraz na dołączonym do instrukcji pliku pdf).



Skrypt tworzący należy wykonać w taki sposób, że w pierwszym etapie zostaną stworzone wszystkie tabele oraz odpowiednie dla nich klucze główne `PRIMARY KEY` a dopiero w drugim etapie zostaną dobudowane (za pomocą polecenia `ALTER TABLE`) wszystkie ograniczenia typu `FOREIGN KEY`. Jeżeli chodzi o ograniczenia typu `UNIQUE` (np. na kolumnie *funkcje.nazwa*), `ENUM` (np. na kolumnie *pracownicy.plec)* oraz `DEFAULT` (np. na kolumnie *pracownicy.zarobki*), to

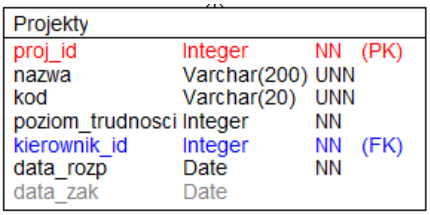
powinny być one utworzone w czasie definiowania tabel jako tzw. ograniczenia tablicowe. Z kolei ograniczenia typu **NOT NULL** definiujemy jako tzw. ograniczenia kolumnowe. Ograniczenia klucza obcego powinny mieć zdefiniowane nazwy (czy potrafisz wyjaśnić dlaczego?)

2. Zwróć uwagę, że w jednym przypadku (którym?) klucz główny tabeli jest kluczem złożonym, opartym o dwie kolumny. Jednocześnie kolumny tworzące klucz główny są też kolumnami tworzącymi klucze obce. Przedyskutuj wady i zalety takiego podejścia.

3. Opisać własnymi słowami utworzony model. Jakie jest jego potencjalne zastosowanie. Do opisu jakiego rzeczywistego problemu został on stworzony?

4. Wstawić do utworzonych tabel przykładowe rekordy: do tabeli *pracownicy* 20 rekordów, do tabeli *projekty* 3 rekordy, do tabeli *funkcje* 5 rekordów, do tabeli *zespoły* 20 rekordów. Wartości dla kolumn będących **PRIMARY KEY** pobierać wykorzystując zdefiniowaną wcześniej opcję **AUTO_INCREMENT**.

5. Zmodyfikować definicję tabeli projekty (polecenie **ALTER**). Dodać kolumnę **poziom_trudnosci**, tak jak to pokazano na poniższym rysunku.



Kolumna ta powinna mieć ograniczenie CHECK i powinna przyjmować tylko wartości ze zbioru (**łatwy, średni, trudny**). Zwróć uwagę, że kolumna ta powinna pojawić się dokładnie w tym miejscu, jak to pokazano na rysunku (a nie na końcu, jako ostatnia kolumna w tabeli). Ma ona też ograniczenie **NOT NULL**.

6. Na bazie tabel *customer*, *ord*, *item* oraz *product* ze schematu demonstracyjnego zbudować widok (polecenie **CREATE VIEW**) o nazwie *zamowienia_view*. Następujące zapytanie:

```
SELECT * FROM zamowienia_view WHERE razem > 10000;
```

powinno zwrócić wynik jak poniżej:

```
+---------------------------+--------------------+------------+-------+-----------+
| Klient                    | Produkt            | Cena jedn. | Ilosc | Razem     |
+---------------------------+--------------------+------------+-------+-----------+
| Big John's Sports Emporium | Bunny Boot         |     140.00 |   150 |  21000.00 |
| Big John's Sports Emporium | Ace Ski Boot       |     175.00 |   600 | 105000.00 |
| Big John's Sports Emporium | Himalaya Bicycle   |     582.00 |  1500 | 873000.00 |
| Hamada Sport              | Grand Prix Bicycle |    1669.00 |    85 | 141865.00 |
| Kuhn's Sports             | World Cup Net      |     115.00 |   130 |  14950.00 |
| Kuhn's Sports             | Grand Prix Bicycle |    1669.00 |    75 | 125175.00 |
| Muench Sports             | Grand Prix Bicycle |    1669.00 |    19 |  31711.00 |
| Unisports                 | Grand Prix Bicycle |    1500.00 |    50 |  75000.00 |
| Womansport                | Pro Ski Pole       |      36.00 |   400 |  14400.00 |
| Womansport                | Bunny Boot         |     135.00 |   500 |  67500.00 |
| Womansport                | Pro Ski Boot       |     380.00 |   400 | 152000.00 |
| Womansport                | Himalaya Bicycle   |     582.00 |   600 | 349200.00 |
+---------------------------+--------------------+------------+-------+-----------+
12 rows in set, 1 warning (0.00 sec)
```

Struktura widoku jest następująca:

```
mysql> desc zamowienia_view;
+------------+---------------+------+-----+---------+-------+
| Field      | Type          | Null | Key | Default | Extra |
+------------+---------------+------+-----+---------+-------+
| Klient     | varchar(50)   | NO   |     | NULL    |       |
| Produkt    | varchar(50)   | NO   |     | NULL    |       |
| Cena jedn. | decimal(11,2) | YES  |     | NULL    |       |
| Ilosc      | int(11)       | YES  |     | NULL    |       |
| Razem      | decimal(21,2) | YES  |     | NULL    |       |
+------------+---------------+------+-----+---------+-------+
5 rows in set (0.22 sec)
```

7. Zaproponować oraz zbudować 2 inne **sensowne** widoki na bazie schematu demonstracyjnego.