

Tematyka zajęć

Implementacja baz danych: kolumny, typy danych, związki, więzy integralności, sekwencje.

Uwaga: Poniższy zestaw zadań przeznaczony jest na jeden półtoragodzinny blok zajęć oraz pracę samodzielną poza zajęciami laboratoryjnymi. Zadania lub punkty oznaczone symbolem ★ przeznaczone są do realizacji samodzielnej. Podczas zajęć laboratoryjnych należy w pierwszej kolejności realizować pozostałe *obowiązkowe* podpunkty.

Zadanie 9.1

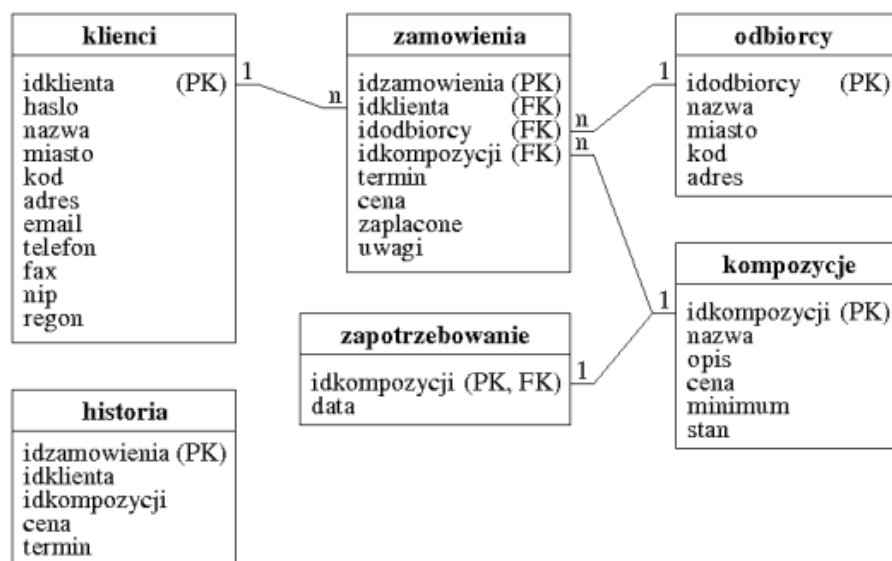
baza danych: kwiaciarnia

Tworzona baza danych ma stanowić podstawę dla istnienia internetowej kwiaciarni. Z jednej strony ma ona umożliwić klientowi wybór jednej lub kilku z gotowych kompozycji z oferty kwiaciarni. Przy złożeniu zamówienia klient musi podać swoje dane (identyfikator i hasło) lub zarejestrować się w systemie jeśli jest nowym klientem (korzysta z kwiaciarni pierwszy raz). Poza tym klient musi określić, gdzie mają dostarczone zostać kwiaty i osobę, której mają one zostać przekazane. Z drugiej strony baza danych ma stanowić podstawę zarządzania kwiaciarnią. Powinny być w niej przechowywane informacje o aktualnych zasobach kwiaciarni (ilości poszczególnych kompozycji) i automatycznie zgłaszane zapotrzebowanie na wyczerpujące się zasoby.

Przygotuj skrypt implementujący bazę danych *kwiaciarnia* zgodnie z przedstawionym poniżej projektem i opisem.

Uwaga: Baza danych kwiaciarnia ma zostać umieszczona w schemacie **kwiaciarnia** (patrz zadanie 1.1).

Uwaga: zapis *maksymalnie n znaków* oznacza, że należy użyć typ `varchar(n)`. Zapis *dokładnie n znaków* oznacza, że należy użyć typ `char(n)`.



- **idklienta** - typ znakowy, maksymalnie 10 znaków (tzn. użyć varchar(10)), klucz główny,
- **haslo** - typ znakowy, maksymalnie 10 znaków, minimum 4 znaki, wymagane,
- **nazwa** - typ znakowy, maksymalnie 40 znaków, wymagane,
- **miasto** - typ znakowy, maksymalnie 40 znaków, wymagane,
- **kod** - typ znakowy, dokładnie 6 znaków, wymagane,
- **adres** - typ znakowy, maksymalnie 40 znaków, wymagane,
- **email** - typ znakowy, maksymalnie 40 znaków,
- **telefon** - typ znakowy, maksymalnie 16 znaków, wymagane,
- **fax** - typ znakowy, maksymalnie 16 znaków,
- **nip** - typ znakowy, dokładnie 13 znaków,
- **regon** - typ znakowy, dokładnie 9 znaków,

- **idkompozycji** - typ znakowy, dokładnie 5 znaków (tzn. użyć char(5)), klucz główny,
- **nazwa** - typ znakowy, maksymalnie 40 znaków, wymagane,
- **opis** - typ znakowy, maksymalnie 100 znaków,
- **cena** - typ numeryczny z dokładnością do 2 cyfr po przecinku, minimalna cena to 40.00 zł,
- **minimum** - typ całkowity,
- **stan** - typ całkowity,

- **idodbiorcy** - typ serial, klucz główny,
- **nazwa** - typ znakowy, maksymalnie 40 znaków, wymagane,
- **miasto** - typ znakowy, maksymalnie 40 znaków, wymagane,
- **kod** - typ znakowy, dokładnie 6 znaków, wymagane,
- **adres** - typ znakowy, maksymalnie 40 znaków, wymagane,

- **idzamowienia** - typ całkowity, klucz główny,
- **idklienta** - typ znakowy, maksymalnie 10 znaków, klucz obcy, wymagane,
- **idodbiorcy** - typ całkowity, klucz obcy, wymagane,
- **idkompozycji** - typ znakowy, dokładnie 5 znaków, klucz obcy, wymagane,
- **termin** - data, wymagane,
- **cena** - typ numeryczny z dokładnością do 2 cyfr po przecinku,
- **zaplacone** - typ logiczny,
- **uwagi** - typ znakowy, maksymalnie 200 znaków,

- **idzamowienia** - typ całkowity, klucz główny,
- **idklienta** - typ znakowy, maksymalnie 10 znaków,
- **idkompozycji** - typ znakowy, dokładnie 5 znaków,
- **cena** - typ numeryczny z dokładnością do 2 cyfr po przecinku,
- **termin** - data,

- **idkompozycji** - typ znakowy, dokładnie 5 znaków, klucz główny i klucz obcy,
- **data** - data.

Uwaga: Sprawdź jaka jest wartość domyślna w kolumnie odbiorcy.idodbiorcy. Skąd się bierze ta wartość?

Zadanie 9.2

baza danych: kwiaciarnia

1. Wykonaj skrypt tworzący relacje w bazie danych *kwiaciarnia*.
2. Przygotuj odpowiednio dane z pliku `kwiaciarnia2dane-tekst.txt` i zaimportuj je do bazy danych.
3. Sprawdź (np. wykonując zapytania), czy wszystkie dane zostały zaimportowane do bazy danych.
4. Jak sprawdzić wartość klucza głównego dla ostatnio dodanego rekordu do tabeli *odbiorcy* w ramach tej samej sesji?

COPY

Dokumentacja: <http://www.postgresql.org/docs/9.1/static/sql-copy.html>

Zwykły użytkownik (nie postgres) może kopiować dane wyłącznie z **stdin**. Zatem w skrypcie przed pierwszym wierszem z danymi kopiowanymi do przykładowej tabeli **abc** należy umieścić polecenie:

```
copy abc from stdin with (... odpowiednie opcje);
```

Po ostatnim wierszu danych kopiowanych do przykładowej tabeli **abc** umieszczamy linię:

```
\.
```

W jednym skrypcie można umieścić kilka poleceń **copy**, kopiujących dane do różnych tabel w bazie danych.

```
copy kwiaciarnia.klienci from stdin with (delimiter ';', null 'BRAK DANYCH');
copy kwiaciarnia.odbiorcy(nazwa,miasto,kod,adres) from stdin with (delimiter ';', null 'BRAK DANYCH');
```

Uwaga: Ze względu na typ **serial** w przypadku tabeli **odbiorcy** konieczne jest wskazanie atrybutów dla których kopiujemy dane.

Zadanie 9.3

baza danych: firma

Przygotuj skrypt implementujący bazę danych *firma* zgodnie z przedstawionym poniżej opisem.

Uwaga: Baza danych ma zostać umieszczona w schemacie **firma**.

Relacja *dzialy* zawiera atrybuty:

- iddzialu - typ znakowy, dokładnie 5 znaków, klucz główny,
- nazwa - typ znakowy, maksymalnie 32 znaki, wymagane,
- lokalizacja - typ znakowy, maksymalnie 24 znaki, wymagane,
- kierownik - liczba całkowita, klucz obcy odwołujący się do pola *idpracownika* w relacji *pracownicy*.

Relacja *pracownicy* zawiera atrybuty:

- idpracownika - liczba całkowita, klucz główny,
- nazwisko - typ znakowy, maksymalnie 32 znaki, wymagane,
- imie - typ znakowy, maksymalnie 16 znaków, wymagane,
- dataUrodzenia - data, wymagane,
- dzial - typ znakowy, dokładnie 5 znaków, wymagane, klucz obcy odwołujący się do pola *iddzialu* w relacji *dzialy*,
- stanowisko - typ znakowy, maksymalnie 24 znaki,
- pobory - typ stałoprzecinkowy z dokładnością do 2 miejsc po przecinku.

Ponieważ występują cykliczne zależności między relacjami *dzialy* i *pracownicy*, klucz obcy w relacji *dzialy* należy dodać za pomocą polecenia *alter table* po utworzeniu obu relacji:

```
alter table dzialy add constraint dzial_fk foreign key(kierownik)
references pracownicy(idpracownika) on update cascade deferrable;
```

Zadanie 9.4

baza danych: firma

1. Wykonaj skrypt tworzący tabele w bazie danych *firma*.
2. ★ Przygotuj skrypt z instrukcjami *insert into*, wstawiający do bazy *firma* poniższe krotki:

```
('PH101','Handlowy','Marki',77)
('PR202','Projektowy','Olecko',27)
('PK101','Personalny','Niwka',72)

(27,'Kruk','Adam','15/12/1989','PK101','kierownik',2200.00)
(270,'Kowalik','Artur','13/12/1988','PK101','analitik',2400.00)
(72,'Kowalik','Adam','15/11/1989','PR202','kierownik',2600.00)
(720,'Kowalik','Amadeusz','17/12/1988','PK101','analitik',3200.00)
(707,'Kukulka','Antoni','15/12/1999','PH101','robotnik',1600.00)
(207,'Kowal','Alojzy','15/10/1998','PH101','robotnik',2000.00)
(77,'Kowalus','Adam','12/11/1998','PH101','kierownik',5400.00)
(1010,'Kawula','Alojzy','15/11/1998','PK101','robotnik',2500.00)
```

Dane stawiamy wewnątrz transakcji:

```
begin;
set constraints dzial_fk deferred;
-- wstawianie danych
commit;
```

★ Zadanie 9.5

baza danych: firma

Napisz zapytanie działające na tabelach bazy danych *firma*, które:

1. Wyświetla nazwisko, wiek oraz roczne pobory pracownika posortowane wg poborów oraz nazwiska (pole pobory w tabeli pracownicy określa pobory miesięczne).
2. Wyświetla nazwisko, imię, datę urodzenia, stanowisko, dział i pobory pracownika, który pracuje na stanowisku robotnik lub analitik i zarabia więcej niż 2000 miesięcznie.
3. Wyświetla nazwiska i imiona pracowników, którzy zarabiają więcej niż zarabia Adam Kowalik.
4. Podnosi zarobki o 10% na stanowisku robotnik.
5. Oblicza średnie zarobki oraz ilość pracowników na poszczególnych stanowiskach z wyłączeniem stanowisk kierownik.