

Nic co się nie kompiluje nie będzie sprawdzane!

W postgresie został utworzony nowy użytkownik o nazwie `<TwojLogin>_other`, który ma prawo łączenia się z Twoją bazą danych; odbywa się to przez wydanie polecenia

`psql <TwojLogin> <TwojLogin>_other`

np. użytkownik `wch` może wydać polecenie `psql wch wch_other`. Poniżej będziemy użytkownika `<TwojLogin>_other` nazywać *drugim użytkownikiem*.

Zadanie 1 (5+5+5+20 pkt).

- (a) Nadaj drugiemu użytkownikowi minimalne prawa wystarczające do wykonania poniższego zadania (w szczególności nie ma najmniejszego powodu nadawania mu prawa czytania tabeli `wybor`). Dalsze punkty tego zadania (o ile nie jest napisane inaczej) mają być wykonane z konta drugiego użytkownika.
- (b) Utwórz tabelę `osoba` zawierającą takie same kolumny jak tabela `uzytkownik` oraz dodatkowo kolumnę `uwagi` typu `text`, kolumnę `mod_id` typu `character varying(15)` i kolumnę `mod_data` typu `timestamp`. Przekopiuj do niej wszystkie dane z tabeli `uzytkownik`.
- (c) Sprawdź czy pierwszy użytkownik ma prawa dostępu do tabeli `osoba`. Jeśli ich nie ma, to nadaj mu wszystkie prawa.
- (d) Napisz wyzwalacz, który przy każdej modyfikacji krotki w tabeli `osoba` wpisze do pól `mod_id` i `mod_data` odpowiednio nazwę użytkownika, który dokonał modyfikacji (do tego przyda się zmienna `user`) oraz dokładny czas tej modyfikacji (przyda się funkcja `now()`).

Zadanie 2 (5+15 pkt).

- (a) Zabierz drugiemu użytkownikowi wszystkie prawa nadane w poprzednim zadaniu. Czy można odebrać mu prawa do tabeli `osoba`?
- (b) Nadaj drugiemu użytkownikowi prawo czytania Twojego planu zajęć (i niczego więcej).
Wskazówka: do tego się używa perspektyw.

Zadanie 3 (20+10 pkt).

- (a) Napisz funkcję `srednia`, która dla podanej nazwy przedmiotu, rodzaju zajęć i nazwy semestru zwróci średnią liczbę osób zapisanych na dany rodzaj zajęć z danego przedmiotu w danym semestrze. Np zapytanie `select srednia('Bazy danych', 'r', 'Semestr letni 2008/2009')`; powinno zwrócić wynik 17.5.
- (b) Podaj wykaz (nazwa przedmiotu, rodzaj zajęć, średnia) średnich obciążeń grup (czyli tego co wylicza funkcja z poprzedniego punktu) w bieżącym semestrze. Uporządkuj wynik malejąco według średnich.

Wskazówka: Podczas wyliczania średniej może być potrzebna konwersja typów (porównaj `select 105::real/6::real`; oraz `select 105/6`;))

Nie stracisz 3 punktów jeśli nie zapomnisz o grupach, do których nikt się nie zapisał.

Zadanie 4 (35 pkt). Napisz funkcję `zmiana()`, która zwiększy o 1 wszystkie kody grup w tabeli `grupa`, nie tracąc przy tym informacji o zapisach do tych grup.

Wskazówka: To można zrobić tak: dla każdej krotki w tabeli `grupa`, rozpoczynając od krotki o największym kodzie, utwórz nową krotkę z tymi samymi danymi i kodem większym o 1; następnie przepnij wszystkie zapisy ze starej grupy do nowej i usuń starą krotkę. To się da zrobić przy użyciu kursorów.