Projekt - System Zarządzania Korporacją X

Dokumentacja

Piotr Kałmuczak

Instrukcja uruchomienia programu

Uruchomienie programu wymaga działania na systemie Linux, posiadania bazy danych PostgreSQL w wersji >= 9.1.23 oraz interpretera języka programowania Python3.

Przed uruchomieniem programu należy stworzyć użytkownika init oraz bazę danych corp zgodnie z modelem fizycznym (plik bdPhysicalModel.sql). Program należy uruchomić komendą python main.py. Pierwsze uruchomienie programu musi być wywołane dodatkowo z parametrem –init. Tworzy ono użytkownika app oraz tablicę employee. Każde kolejne wywołanie powinno zostać wywołane z jednym parametrem – plikiem w formacie JSON, z którego czytane będzie wejście. W folderze znajduję się również plik restore.dmp. W przypadku, kiedy będziemy chcieli wrócić do stanu sprzed pierwszego uruchomienia programu z parametrem –init, należy wywołać ten skrypt w bazie danych Postgres za pomocą komendy: \i restore.dmp

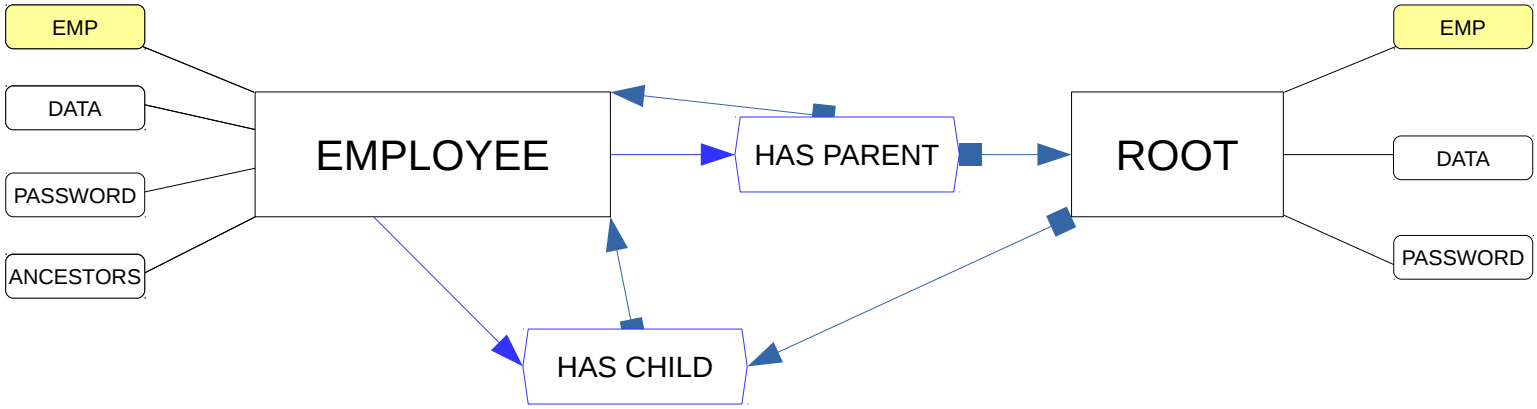
Format pliku wejściowego

Każda linia pliku wejściowego zawiera obiekt JSON (http://www.json.org/json-pl.html). Każdy z obiektów opisuje wywołanie jednej funkcji API wraz z argumentami.

Przykład: obiekt { "function": { "arg1": "val1", "arg2": "val2" } } oznacza wywołanie funkcji o nazwie function z argumentem arg1 przyjmującym wartość "val1" oraz arg2 – "val2".

W pierwszej linii wejścia znajduje się wywołanie funkcji open z argumentami umożliwiającymi nawiązanie połączenia z bazą danych.

DIAGRAM E-R



ROLA EMPOYEE:

- posiada jednego bezpośredniego przełożonego (może posiadać wielu pośrednich),
- może posiadać jednego lub więcej podwładnych (lub nie posiada ich wcale),
- może usuwać swoich bezpośrednich lub pośrednich podwładnych (usunięcie podwładnego powoduję również usunięcie wszystkich bezpośrednich i pośrednik podwładnych usuwanego podwładnego);

ROLA BOSS, tak jak EMPLOYEE, ale:

- nie posiada przełożonego,
- wszyscy są jego przełożonymi,
- nie można go usunąć,
- jest tylko jeden prezes (boss).

Inne komenatrze:

- lista ANCESTORS w tabeli EMPLOYEE, to lista maksymalnie 6-elementowa, która zwiera kolejnych przełożonych (ich klucze) począwszy od roota (bossa), w szczególności ANCESTORS[0] zawsze jest rootem,
- Relacja HAS PARENT po wyjściowej stronie odnosi się albo do EMPLOYEE albo do ROOT.

Opis operacji:

- new <admin><passwd><data><newpasswd><emp1><emp>
  - dodajemy nową krotkę [<emp><data><newpasswd><ancestors>], gdzie <ancestors> są dziedziczeni po <ancesotrs> z <emp1> oraz dodajemy <emp1> na koniec
- remove <admin><passwd><emp>
  - usuwamy pracownika z kluczem <emp> wraz bezpośrednimi i pośrednimi przełożonymi <emp>
- child <admin><passwd><emp>
  - na podstawie <ancestors> dla pracownika i kluczu <emp> ustalamy jego głębokość i w hierarchii, a następni zwracamy tabelę pracowników, którzy na pozycji i+1 w <ancestors> posiadają klucz <emp>
- parent <admin><passwd><emp>
  - zwraca pracownika, który jest jako ostatni na liście <ancestors> pracownika i kluczu <emp>
- ancestors <admin><passwd><emp>
  - tak jak child, ale szukamy klucza <emp> na pozycjach k > i w <ancestors>
- ancestor<admin><passwd><emp1><emp2>
  - sprawdzamy, czy na liście <ancestors> pracownika o kluczu <emp1> widnieje klucz <emp2>
- read <admin><passwd><emp>
  - zwracamy <data> pracownika i kluczu <emp>
- update <admin><passwd><emp><newdata>
  - zmieniamy <data> pracownika o kluczu <emp> na <newdata>