Julia Ruszer 247775

Dominik Gałkowski 247659

Jan Śladowski 247806

Wiktor Żelechowski 247833

Microsoft SQL Server

Zadanie 1.

**1. Utwórz nowe loginy o nazwach login1, login2 i login3 z domyślną bazą danych HR.**

create login login1 with password = 'login1', DEFAULT\_DATABASE = HR;

create login login2 with password = 'login2', DEFAULT\_DATABASE = HR;

create login login3 with password = 'login3', DEFAULT\_DATABASE = HR;

**2. Utwórz nowych użytkowników w bazie danych HR o nazwach user1, user2 i user3 przypisanych do loginów login1, login2 i login3.**

create user user1 for login login1;

create user user2 for login login2;

create user user3 for login login3;

**3. Nadaj użytkownikom user1, user2 i user3 uprawnienia do wyświetlania departamentów.**

grant select on departments to user1, user2, user3;

**4. Utwórz nową rolę o nazwie role1 i uczyń użytkownika user1 jej właścicielem.**

create role role1 authorization user1;

**5. Nadaj użytkownikowi user1 uprawnienia do dodawania departamentów z możliwością dalszego przekazywania uprawnień.**

grant insert on departments to user1 with grant option;

**6. Nadaj roli role1 uprawnienia do usuwania departamentów.**

grant delete on departments to role1;

**7. Jako user1 nadaj użytkownikowi user2 uprawnienia do dodawania departamentów.**

execute as user = 'user1';

grant insert on departments to user2;

**8. Jako user1 nadaj użytkownikowi user2 rolę role1.**

alter role role1 add member user2;

**9. Pozbaw użytkownika user1 uprawnień do dodawania departamentów.**

revert;

revoke insert on departments to user1 cascade;

**10. Pozbaw użytkownika user1 roli role1.**

exec sp\_droprolemember 'role1', 'user1';

**11. Czy jako użytkownik user2 możesz nadać użytkownikowi user3 uprawnienia do dodawania departamentów?**

execute as user = 'user2';

grant insert on departments to user3;

--nie mozna

**12. Czy jako użytkownik user2 możesz nadać użytkownikowi user3 rolę role1?**

alter role role1 add member user3;

--nie mozna

**13. Czy jako użytkownik user1 możesz nadać użytkownikowi user3 rolę role1?**

revert;

execute as user = 'user1';

alter role role1 add member user3;

--mozna

Zadanie 2.

**1. Utwórz tabelę o nazwie admin\_logs, która zawierać będzie następujące pola:**

* **log\_id typu numerycznego będące kluczem głównym,**
* **log\_desc typu znakowego,**
* **log\_data typu datowego (z uwzględnieniem czasu).**

CREATE TABLE admin\_logs ( log\_id INT IDENTITY(1,1) PRIMARY KEY, log\_desc NVARCHAR(255), log\_data DATETIME );

**2. Utwórz nowe zadanie o nazwie login\_report, które:**

* **przy każdym uruchomieniu doda do tabeli admin\_logs wpis zawierający liczbę wszystkich loginów,**
* **począwszy od 1 grudnia 2024 roku będzie wykonywane codziennie o godzinie 2:00.**

CREATE PROCEDURE insert\_login\_report

AS

BEGIN

INSERT INTO admin\_logs (log\_desc, log\_data)

SELECT CONCAT('Liczba loginów: ', COUNT(\*)), GETDATE()

FROM sys.syslogins;

END;

**Obraz zawierający tekst, zrzut ekranu, oprogramowanie, wyświetlacz

Opis wygenerowany automatycznie**

**Obraz zawierający tekst, zrzut ekranu, oprogramowanie, wyświetlacz

Opis wygenerowany automatycznie**

**Obraz zawierający tekst, zrzut ekranu, oprogramowanie, numer

Opis wygenerowany automatycznie**

**Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie**

PostgreSQL

Zadanie 1.

**1. Utwórz nowych użytkowników o nazwach user1, user2 i user3.** = HR;

create user user1;

create user user2;

create user user3;

**2 Nadaj użytkownikom user1, user2 i user3 uprawnienia do wyświetlania departamentów.**

grant select on table departments to user1, user2, user3;

**3. Utwórz nową rolę o nazwie role1 i uczyń użytkownika user1 jej właścicielem.**

create role role1 with admin user1;

**4. Nadaj użytkownikowi user1 uprawnienia do dodawania departamentów z możliwością dalszego przekazywania uprawnień.**

grant insert on table departments to user1 with grant option;

--grant option umozliwia nadanie przywileju obiektowego innym uzytkownikom (z grant lub bez)

--lub rolom

**5. Nadaj roli role1 uprawnienia do usuwania departamentów.**

grant delete on table departments to role1;

**6. Jako user1 nadaj użytkownikowi user2 uprawnienia do dodawania departamentów.**

set role user1;

grant insert on table departments to user2;

**7. Jako user1 nadaj użytkownikowi user2 rolę role1.**

grant role1 to user2;

**8. Pozbaw użytkownika user1 uprawnień do dodawania departamentów.**

reset role;

revoke insert on table departments from user1 CASCADE;

**9. Pozbaw użytkownika user1 roli role1.**

revoke role1 from user1 CASCADE;

**10. Czy jako użytkownik user2 możesz nadać użytkownikowi user3 uprawnienia do dodawania departamentów?**

set role user2;

grant insert on table departments to user3;

--nie mozna

**11. Czy jako użytkownik user2 możesz nadać użytkownikowi user3 rolę role1?**

grant role1 to user3;

--nie mozna

**12. Czy jako użytkownik user1 możesz nadać użytkownikowi user3 rolę role1?**

reset role;

set role user1;

grant role1 to user3;

--nie mozna

Zadanie 2.

**1. Utwórz tabelę o nazwie admin\_logs, która zawierać będzie następujące pola:**

* **log\_id typu numerycznego będące kluczem głównym,**
* **log\_desc typu znakowego,**
* **log\_data typu datowego (z uwzględnieniem czasu).**

**create table admin\_logs(**

**log\_id serial primary key,**

**log\_desc varchar,**

**log\_data timestamp**

**)**

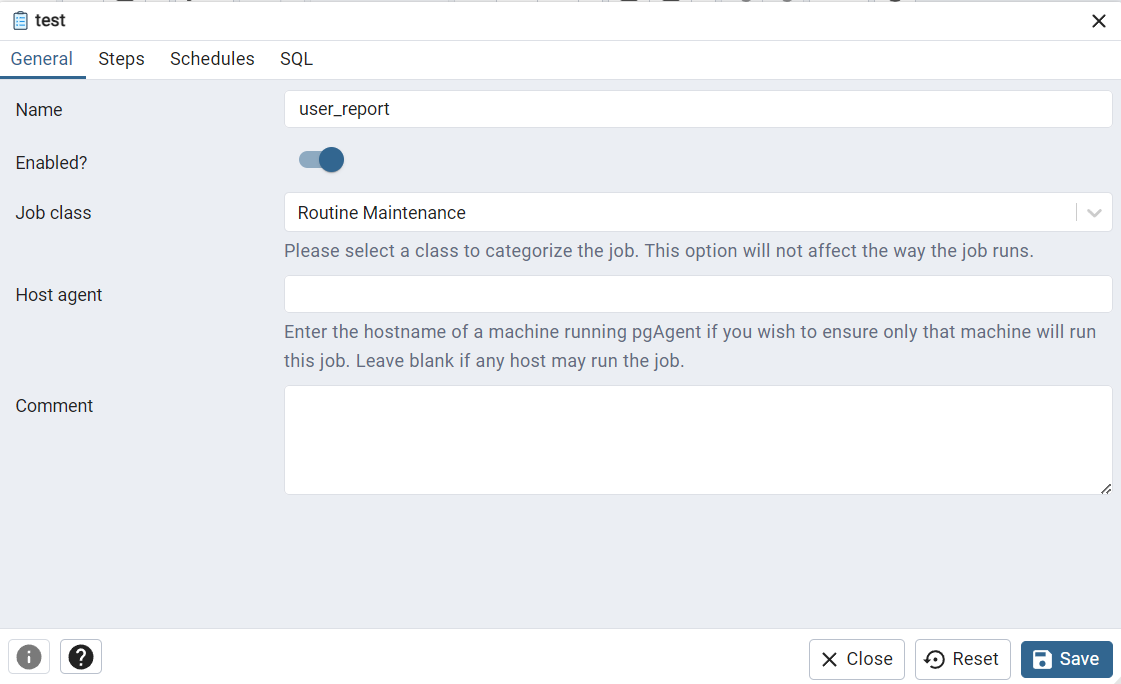
**2. Utwórz nowe zadanie o nazwie user\_report, które:**

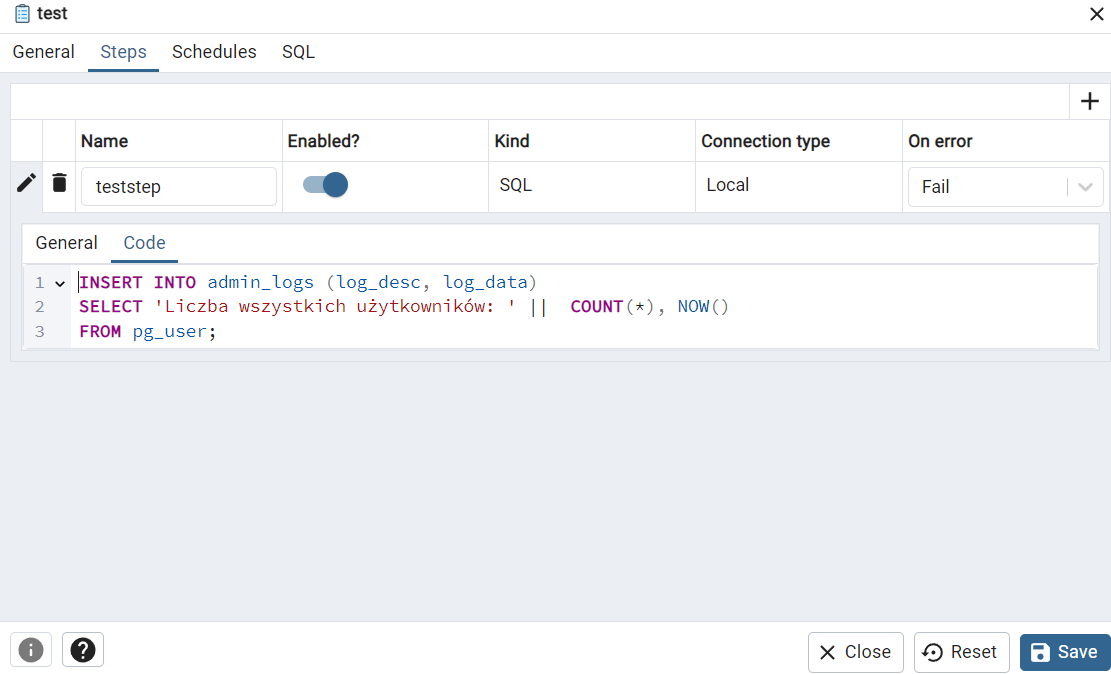
* **przy każdym uruchomieniu doda do tabeli admin\_logs wpis zawierający liczbę wszystkich użytkowników,**
* **począwszy od 1 grudnia 2024 roku będzie wykonywane codziennie o godzinie 2:00.**

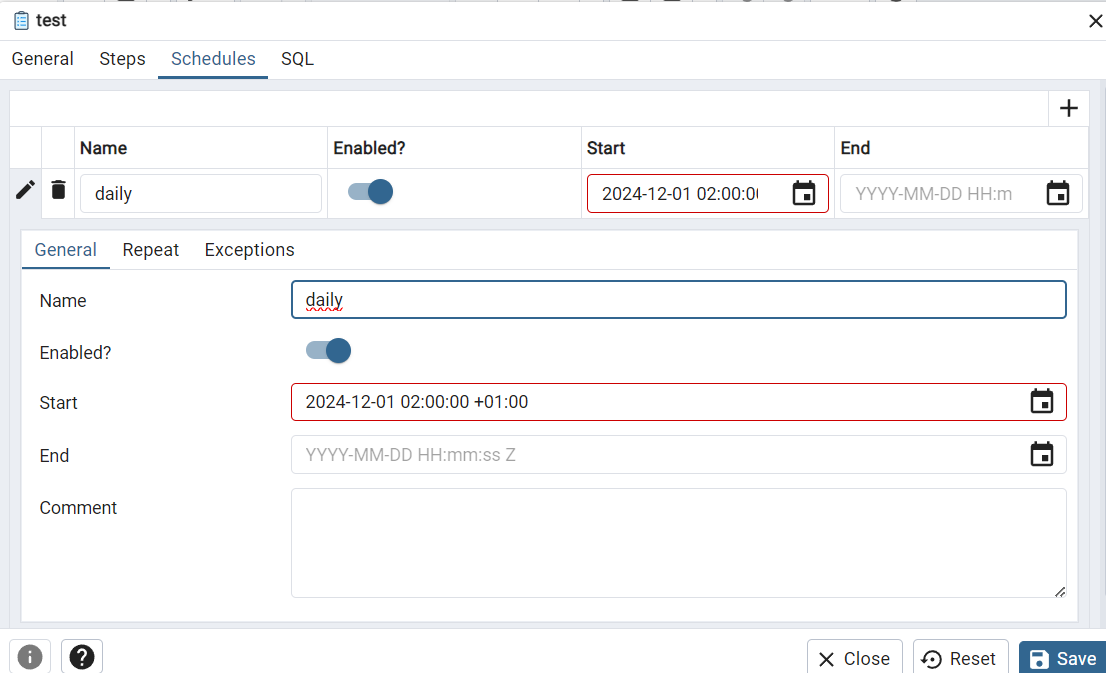
**INSERT INTO admin\_logs (log\_desc, log\_data)**

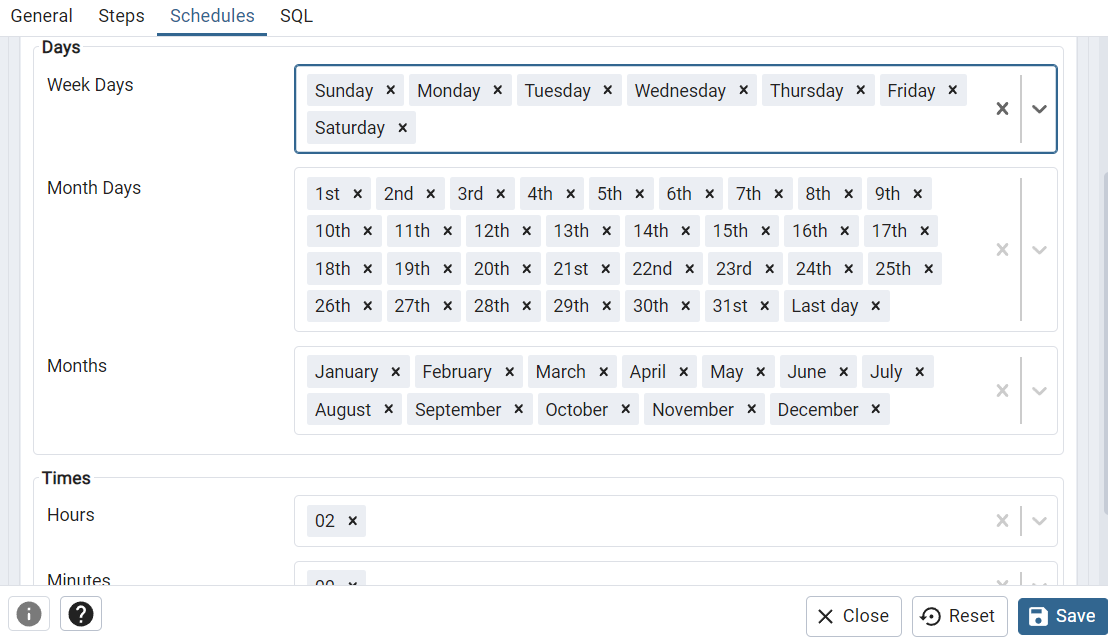
**SELECT 'Liczba wszystkich użytkowników: ' || COUNT(\*), NOW()**

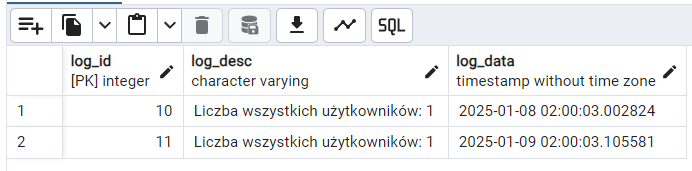
**FROM pg\_user;**

****

****

****

****

****