

Домашнее задание "Консоль. Функции. Триггеры"

Задание 1. World-db. (см скрин 1)

Для командной строки (cmd) восстановление бэкапа будет выглядеть так (путь до файла *pg_restore.exe* указан в переменных винды) (все команды в файле DZ_3-1_Restore_backup.bat):

```
pg_restore.exe --host localhost --port 5432 --username postgres --dbname netology --verbose "d:\YandexDisk\_Pentaho_IN\world-db.backup
```

Полный путь выглядит так:

```
"c:\Program Files\PostgreSQL\13\bin\pg_restore.exe" --host localhost --port 5432 --username postgres --dbname netology --verbose  
"d:\YandexDisk\_Pentaho_IN\world-db.backup
```

Выполнить запрос в cmd команда:

```
psql --host localhost --port 5432 --username postgres --dbname netology -c "select * from world.country;"
```

Но получаю ошибку: ОШИБКА: для символа с последовательностью байт 0xc3 0xab из кодировки "UTF8" нет эквивалента в "WIN1251"

Получилось получить 3 первых записи :

```
psql --host localhost --port 5432 --username postgres --dbname netology -c "select * from world.country limit 3;"
```

В четвертой есть символы для которых нет эквивалента в "WIN1251"

Поменять кодировку на UTF8 в cmd у меня так и не вышло:

Получить текущую кодировку:

```
psql --host localhost --port 5432 --username postgres --dbname netology -c "\encoding"
```

WIN1251

Команды вида:

```
psql --host localhost --port 5432 --username postgres --dbname netology -c "\set client_encoding 'utf8';"
```

ошибок не выдавали, но при этом запрос отрабатывает с ошибкой, указанной выше.

Для SQL Shell (psql) запрос отрабатывает без ошибок (см скрин 2):

```
set client_encoding='utf8';
```

```
select * from world.country;
```

Задание 2. Прикладываю sql файл:

-- функция, в виде аргументов принимает две даты и в качестве
-- результата возвращает сумму продаж между этими датами, включая эти даты.

-- ВАРИАНТ 1

```
create or replace function sales_amount_period(d1 date, d2 date, out sum_sales numeric(9,2))
as $$
begin
    d2 = d2 + 1;
    select
        sum(f.rental_rate)
    from
        rental r
    into sum_sales
    left join inventory i on i.inventory_id = r.inventory_id
    left join film f on f.film_id = i.film_id
    where
        Rental_date between d1 and d2;
    return;
end $$
language plpgsql;

select sales_amount_period('2005/05/24', '2005/05/24'); -- 19.92
select sales_amount_period('2005/05/24', '2005/05/25'); -- 443,55
```

-- ВАРИАНТ 2

```
create or replace function sales_amount_period_2(d1 date, d2 date) returns numeric(9,2)
as $$
declare
    sum_sales numeric(9,2);
begin
    d2 = d2 + 1;
    select
        sum(f.rental_rate)
    from
        rental r
    into sum_sales
    left join inventory i on i.inventory_id = r.inventory_id
    left join film f on f.film_id = i.film_id
    where
        Rental_date between d1 and d2;
    return sum_sales;
end $$
language plpgsql;
```

```
-- проверка
select sales_amount_period_2('2005/05/24', '2005/05/24');      -- 19.92
select sales_amount_period_2('2005/05/24', '2005/05/25');      -- 443,55
```

Задание 3. Прикладываю sql файл:

```
-- Создание таблицы not_active_customer
-- DROP TABLE public.not_active_customer;

CREATE TABLE public.not_active_customer (
    id serial NOT null primary key,
    -- customer_id int2 NOT NULL,
    customer_id int2 NOT NULL references public.customer(customer_id),
    not_active_date date not NULL DEFAULT now()::date
);

-- Создание триггерной функции
create or replace function f_not_active_customer() returns trigger as $$
begin
    if old.active=1 and new.active=0 and TG_OP = 'UPDATE' then
        -- insert into public.not_active_customer select new.customer_id;
        INSERT INTO public.not_active_customer(customer_id) VALUES(NEW.customer_id);
    end if;
    return null;
end;
$$ language plpgsql;

-- создание триггера
drop trigger if exists audit_active_customer on public.customer;
create trigger audit_active_customer
after update on public.customer
for each row execute function f_not_active_customer(customer_id);

-- проверка:
update public.customer SET active = 0 WHERE customer_id < 21;
```