practica 6

1-

a-

Te trae todas las columnas de la tabla sitio.

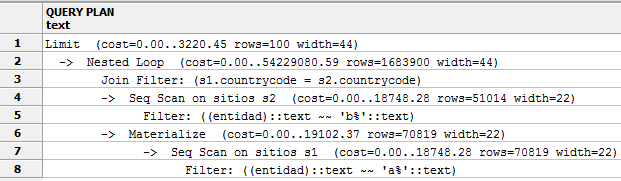
Para countrycode = countrycode en dos instancias de la tabla.

y el campo entidad empieza en la primera instancia con a y en la segunda con b.

Límite de 100.

Había traído todas las de brazil.

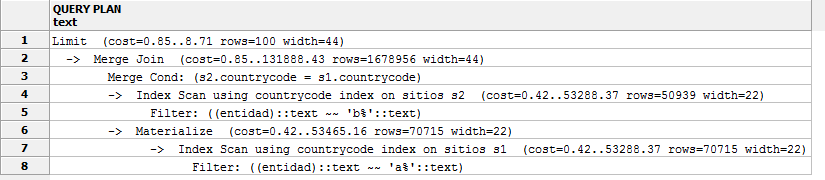
b-



c-

Trae de diferentes países después de haber creado el índice. (comenzando por países con A).

d-



2-

CREATE VIEW vistap2 AS

select a.name, a.population as populationTotal, sum(b.population) as populationCity, (sum(b.population)\*100/a.population) as porcentaje

from country a inner join city b on a.code=b.countrycode

group by a.population, a.name

order by porcentaje desc

3-

a-

CREATE TABLE sitios\_audit(

operacion char(1) NOT NULL,

fecha timestamp NOT NULL,

userid text NOT NULL,

id integer NOT NULL,

entidad character varying NOT NULL,

tipo\_entidad character varying NOT NULL,

countrycode character(3)

);

CREATE OR REPLACE FUNCTION process\_sitios\_audit() RETURNS TRIGGER AS $sitios\_audit$

BEGIN

--

IF (TG\_OP = 'DELETE') THEN -- TG\_OP ES LO QUE TE DICE LA OPERACION

INSERT INTO sitios\_audit SELECT 'D', now(), user, OLD.\*; -- OLD ES RECORD VIEJO DE LO QUE HABIA EN ROW

RETURN OLD;

ELSIF (TG\_OP = 'UPDATE') THEN

INSERT INTO sitios\_audit SELECT 'U', now(), user, NEW.\*;

RETURN NEW;

ELSIF (TG\_OP = 'INSERT') THEN

INSERT INTO sitios\_audit SELECT 'I', now(), user, NEW.\*;

RETURN NEW;

END IF;

RETURN NULL; -- es un trigger AFTER entonces ignoramos resultados

END;

$sitios\_audit$ LANGUAGE plpgsql;

CREATE TRIGGER sitios\_audit

AFTER INSERT OR UPDATE OR DELETE ON sitios

FOR EACH ROW EXECUTE PROCEDURE process\_sitios\_audit();

b-

INSERT INTO sitios (id,entidad,tipo\_entidad,countrycode) VALUES (1234567,'prueba','pru','ARG')

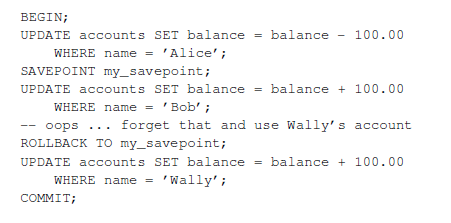
INSERT INTO sitios (id,entidad,tipo\_entidad,countrycode) VALUES (1234568,'prueba2','pru2','ARG')

DELETE FROM sitios

WHERE id = 1234567

4

a-



b-

Una transacción es esencial porque agrupa múltiples pasos (múltiples queries) en un paquete que ejecuta todo o no ejecuta nada. Esto se hace en caso de que falle el sistema en el medio de una serie de queries que es imperativo que se ejecuten todas.

Ejemplo del banco en el que se debitan 100$ de mi cuenta, y deberían ir a la cuenta de Camila pero se cae el sistema después de debitarme y nunca le llega a Camila.

La idea es que se debite y se acredite todo en una misma transacción o no se debite ni se acredite nada.

A su vez se pueden crear puntos de guardado (eso verlo mejor) y hacer roll backs para volver para atrás en caso de ser necesario.

5-

CREATE OR REPLACE FUNCTION avgcities(code character(3)) RETURNS numeric AS $$

DECLARE

prom numeric;

BEGIN

SELECT AVG (population) into prom FROM city WHERE countrycode = code;

RETURN prom;

END;

$$ LANGUAGE plpgsql;

select "avgcities"('ARG');