TRABAJO PRÁCTICO

BASE DE DATOS 2 - POSTGRES SQL

Alumno Bruno: Bidone

Docente: Ignacio Bisso

Cátedra 01

Universidad Nacional de General Sarmiento

Fecha de entrega: sábado 22 de abril de 2017

# Actividad

1 Armar un shell script llamado carga\_inicial.sh que cree las tablas ALUMNO, MATERIA y CURSA y les cargue 10000 alumnos, 20 materias y entre 0 y 4 inscripciones por alumno al azar respectivamente. El script debe borrar las tablas si existen previamente (ver DROP IF EXISTS).

2 Armar un shell script con SQL embebido llamado query1.sh que ejecute el query de la guia de laboratorio que devuelve los nombres de los alumnos que cursan 3 materias usando un subquery correlacionado

3 Armar un shell script con SQL embebido lamado query2.sh que ejecute el query de la guia de laboratorio que devuelve los nombres de los alumnos que cursan 3 materias sin usar un subquery correlacionado

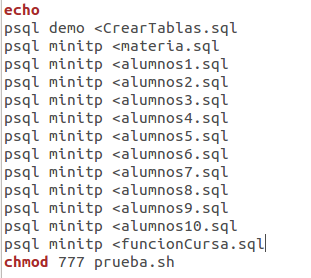
4 Ejecutar los scripts de 2 y 3 anteponiendo el comando time para medir el tiempo de ejecución, comparar y explicar brevemente la causa de la duración de cada query.

5 Es posible acelerar alguno de los queries ? En caso de ser posible implementar la mejora en un tercer shell script (query3.sh) y explicar cómo se logró la mejora y el tiempo resultante.

Tiempo de query1.sh: **252 segundos (tiempo real)**

Tiempo de query2.sh: **151 segundos (tiempo real)**

# Resolución

1. Código carga\_inicial.sh  
     
     
     
   En este script se ejecutan los queries para crear las tablas (alumno, materia y cursa), luego el insert de materia, los insert para los 1000 alumnos y por ultimo funcionCursa que se ocupa de llenar la tabla cursa con la restricción de entre 0 y 4.

A) CrearTablas.sql  
DROP DATABASE IF EXISTS MiniTp;

CREATE DATABASE MiniTp

WITH OWNER = demo

ENCODING = 'UTF8'

TABLESPACE = pg\_default

LC\_COLLATE = 'es\_AR.UTF-8'

LC\_CTYPE = 'es\_AR.UTF-8'

CONNECTION LIMIT = -1;

\c minitp

CREATE TABLE public.alumno

(

legajo serial,

nombre character varying(15) NOT NULL,

apellido character varying(15) NOT NULL,

dni character varying(15),

f\_nacimiento date,

CONSTRAINT alumno\_pkey PRIMARY KEY (legajo)

)

WITH (

OIDS=FALSE

);

ALTER TABLE public.alumno

OWNER TO demo;

CREATE TABLE public.materia

(

nombre character varying(50),

cantidad\_horas integer,

promocionable boolean,

codigo character varying(7) NOT NULL,

CONSTRAINT materia\_pkey PRIMARY KEY (codigo)

)

WITH (

OIDS=FALSE

);

ALTER TABLE public.materia

OWNER TO demo;

CREATE TABLE public.cursa

(

legajo integer NOT NULL,

inscripcion\_ts character varying(20),

codigo character varying(7) NOT NULL,

CONSTRAINT cursa\_codigo\_fkey FOREIGN KEY (codigo)

REFERENCES public.materia (codigo) MATCH SIMPLE

ON UPDATE NO ACTION ON DELETE NO ACTION,

CONSTRAINT cursa\_legajo\_fkey FOREIGN KEY (legajo)

REFERENCES public.alumno (legajo) MATCH SIMPLE

ON UPDATE NO ACTION ON DELETE NO ACTION

)

WITH (

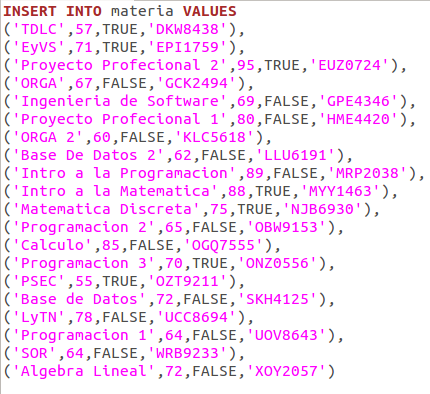
OIDS=FALSE

);

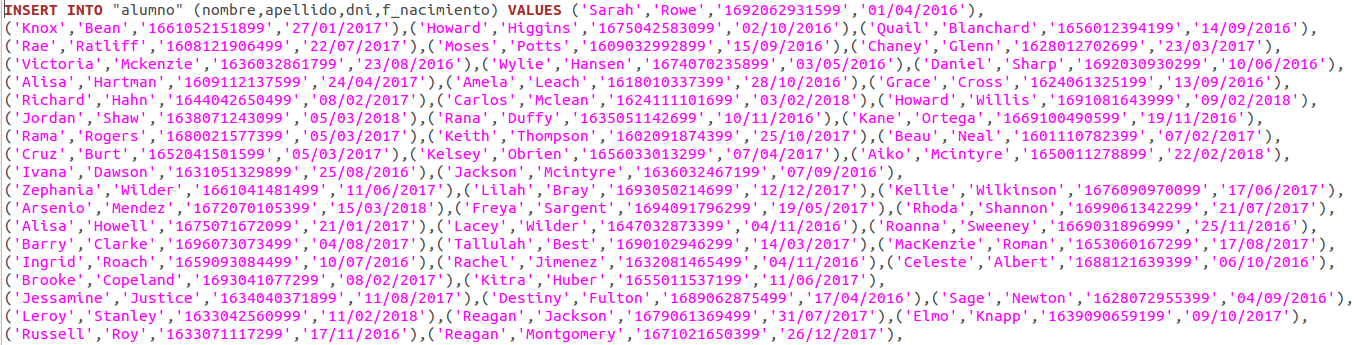
ALTER TABLE public.cursa

OWNER TO demo;

### B) materia.sql

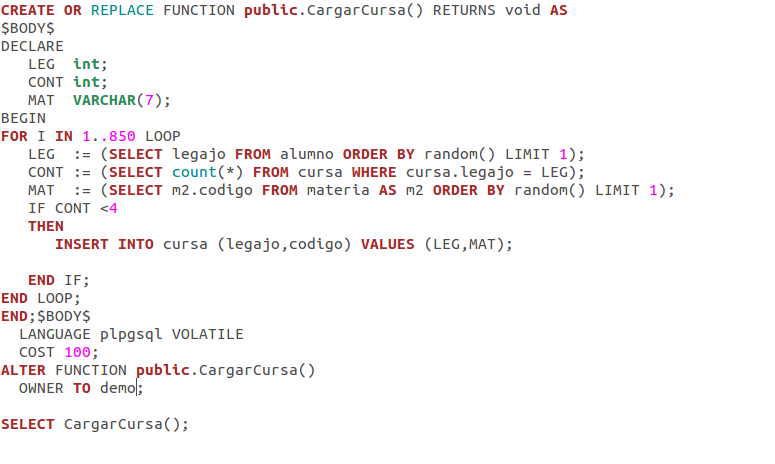


### C) alumnos1.sql

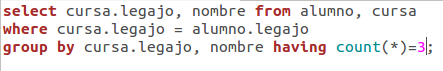


Se ejecutan 10 archivos de alumnos para poder realizar la carga de los 1000 alumnos, en este caso mostramos un pedazo del código utilizado.

### D) funcioncursa.sql



Para llenar la tabla cursa utilizamos la función que se ve en la imagen de arriba, la cual valida que la cantidad de materias cursadas por alumnos no sea mayo a cuatro.

1. Query Correlacionado  
   
2. Query no Correlacionado  
   
3. TIEMPOS  
   Query1.sh 252 segundos en tiempo real.  
   Query2.sh 151 segundos en tiempo real.
4. Índices

En la mayoría de los casos es posible mejorar la performance de los query utilizando índices. Existen tres tipos de índices, I\_primary se crea automáticamente cuando se establece un campo como clave primaria. I\_comun lo valores o necesariamente son únicos y acepta valores null, puede haber varios por tabla. I\_unico, los valores deben ser únicos y diferentes. Permite valores nulos y pueden definirse varios por tabla.

Podríamos tomar este último tipo de índice y aplicarlo a nuestra tabla cursa y crear un índice único por los campos legajo y código.