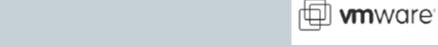
CURSO DE FUNDAMENTOS Y ADMINISTRACION DE DBMS POSTGRES

UBUNTU 10.4

















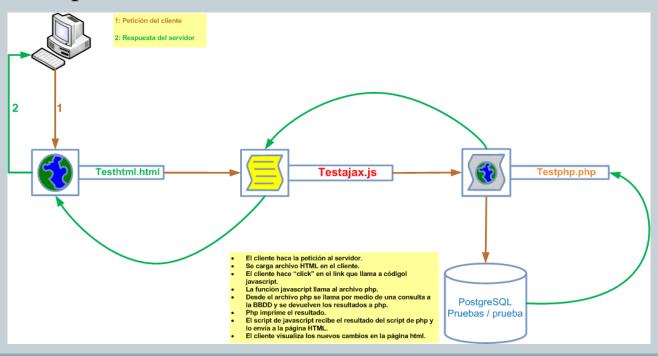




INSTRUCTOR: RUDY SALVATIERRA RODRIGUEZ

POSTGRES:PRIMEROS PASOS

- Que es PostgreSQL?
 - Sistema de gestión de bases de datos.
 - Licencia BSD (Berkeley SoftwareDistribution)
 - Parecida OpenSSLo la MIT License



POSTGRES: CARACTERISTICAS

• Características de PostgreSQL

- Object-Relational, en PostgreSQL cada tabla es definida como una clase, puede implementar la herencia
- O Uso de estándares SQL92, SQL99, etc.
- Código abierto, es una de RDBMS de código abierto mas grande en la actualidad.
- Procesamiento de transacciones
- Integridad referencial
- Herencia de tablas.
- Sistema de reglas.
- o varios lenguajes procedurales

POSTGRES: CARACTERISTICAS

Características de PostgreSQL

- Varios lenguajes procedurales
- Índices compuestos, únicos, parciales, funcionales (sobre funciones).
- o Escritura adelantada de registros (WAL)
- O Juegos de caracteres internacionales.
- O Características para la integridad de los datos,.
- Interfaces de programación para Java (JDBC), ODBC,
 Perl, Python, Ruby,C, C++, PHP, Lisp, Scheme, y Qtsolo para nombrar algunas

POSTGRES:TERMINOLOGIA

- Database
- Query
- Table(relacion, File, Clase)
- ORow
- View
- OClient

- Server
- Postmaster
- Transaction
- Commit
- ORollback
- Index
- Result set

POSTGRES: COMPARACION

Sistema	MySQL	PostgreSQL	SAP DB
Versión	Mysql-3.23.41	PostgreSQL 7.1.3	SAP DB Version 7.3
Licencia	GPL	BSD	GPL
Cumplimiento con	Media	Alta	-
estándar SQL			
Velocidad	Media/Alta	Media	-
Estabilidad	Alta / Muy Alta	Alta	-
Integridad de datos	NO	Si	Si
Seguridad	Alta	Media	-
Soporte de LOCKING	Media	Alta	-
y CONCURRENCIA			
Soporte de Vistas	No (Planeada v4.2)	Si	Si
Soporte Subconsultas	No (Planeada v4.1)	Si	Si
Replicacion	Si	Si	-
Procedimientos	No	Si	Si
almacenados			
Soporte Unicode	NO	Si	-
Soporte Disparadores	No	Si	Si
Integridad referencial	No	Si	Si
	ODBC, JDBC, C/C++,	ODBC, JDBC, C/C++,	ODBC ,JDBC,
Interfaces de	OLEDB, Delphi, Perl,	SQL embebido (en C),	C/C++, Precompilado(SQL
programación	Python, PHP	Tcl/Tk,Perl, Python,	Embebido), Perl, Python,
		PHP	PHP
Tipos de	ISAM, MYISAM,	PostgreSQL mantiene	
Tablas	BerkeleyDB,	su propio sistema	-
alternativas	InnoDB, HEAP,	de tipos	
	MERGE, Gemini	de tablas	
Transacciones	si	Si	-
Claves foráneas	NO (Planeado v4.0)	Si	-
Backups	Si	Si	-
en caliente			

POSTGRES:INSTALACION

Prerrequisitos de software

postgresql

postgresql-server

postgresql-devel

postgresql-odbc

o postgresql-jdbc

postgresql-test

postgresql-libs

postgresql-docs

postgresql-contrib

Clients, libraries, and documentation

Programs (and data files) required to

run a server

required to create new client applications

ODBC driver for PostgreSQL

JDBC driver for PostgreSQL

Regression test suite for PostgreSQL

Shared libraries for client applications

Extra documentation not included in

the postgresql base package

Contributed software

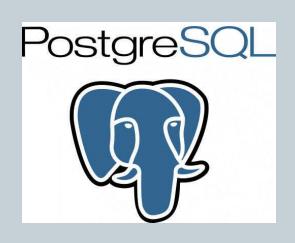
POSTGRES:INSTALACION

Para la instalación de sistemas operativos Linux se deben considerar los siguientes paquetes.

- postgresql
- postgresql-client
- postgresql-contrib
- libpq-dev
- opgadmin3

POSTGRES:INSTALACION

• La guía de instalación la encontrara en la carpeta videos instalación de Postgresql server en el cual encontrara como instalar Postgresql Server, Servidor Xampp y el cliente Ems postgresql que se utilizara en este curso y la instalación en Linux Ubuntu.









POSTGRES: PRUEBAS DE CONECCION

- Comando para conexión a una Base de Datos
 - \$psql -d prueba
- \$psql --help
 Para mostrar todas la opciones

Las opciones mas importantes

- o -d o -W<BD> Base de datos al que se desea conectarse.
- o -f <archivo> Cargar consultas desde un archivo script
- o -h <Host> Servidor de BD al que se desea conectar
- o -p <pue>puerto> Puerto del servidor de BDs
- o -U <user> Usuario de la BD con el que se quiere conectar

POSTGRES: PRUEBAS DE CONECCION

 Para poder conectarnos pos cliente consola tendremos que ubicarnos en la siguiente dirección en el sistema.

C:\Program Files (x86)\PostgreSQL\8.4\bin>_

• Luego si quisiéramos ingresar con este cliente colocamos la siguiente sentencia:

C:\Program Files (x86)\PostgreSQL\8.4\bin>psql -h localhost -U postgres -d postgres
res

Host
Usuario
Base de datos

POSTGRES: PRUEBAS DE CONECCION

• Ejemplos para conexión a una Base de Datos

- \$psql
 Conexión local la mas simple
- \$psql -d prueba
 Conexión local a la BD prueba
- \$psql –U juan -d prueba Conexión local como usuario Juan
- \$psql –U juan –W -d prueba Conexión local y le pide mas el promtp del password
- \$psql -h 192.168.2.11 -U juan -d prueba Conexión remota al servidor de BDs 192.168.2.11 como usuario Juan
- \$psql −192.168.2.11 -U juan −p 5432 -d prueba Conexión remota al servidor de BDs por el puerto estándar 5432

POSTGRES:SENTENCIAS

• Interacción con la BD después de la Conexión

- o prueba=> Promtp que indica conexión a la BD prueba
- o prueba=>\? Muestra opciones de ayuda, entre las mas importantes:

\c <BD> Permite conectar a otra base de datos

\d Muestra descripción de la tabla dada

 $\d{t|i|s|v}$ Lista tablas, índices, secuencias y vistas

respectivamente

\df Lista funciones

POSTGRES:SENTENCIAS

- Interacción con la BD después de la Conexión
 - Lista todas las BDs del servidor
 - \i \archivo \ Ejecuta consultas desde un archivo, similar a la opción '-f' del comando psql.
 - Muestra los permisos respecto a una tabla.
 - \h < cmd SQL> Muestra ayuda en la sintaxis de un comando SQL.
 - Salir de la BD

POSTGRES:LINUX

- Antes, crear una base de datos y usuario en el Servidor local
 - 1. Conectarse como root a la maquina local #su postgres
 - 2. Conectarse con la base de datos template1 \$psql -d template1
 - 3. Crear una base de datos dentro de template1 template1=#create database prueba;
 - 4. Crear un usuario llamado prueba con password prueba template1=#create user prueba with encrypted password 'prueba';
 - 6. permite salir template1=#\q
 - 7. para cambiar de usuario \$exit

 Después de ingresar al servidor de postgres creamos el usuario prueba con password .prueba. ,de la siguiente manera

CREATE USER PRUEBA WITH ENCRYPTED PASSWORD 'PRUEBA';

• Luego creamos la base de datos llamada prueba

CREATE DATABASE PRUEBA;

Para crear usuarios existen tipos de privilegios como por ejemplo.

- SUPERUSER > privilegio de súper usuario
- CREATEDB > permiso para crear bases de datos
- CREATE ROLE > permiso para crear usuarios

Ejemplo

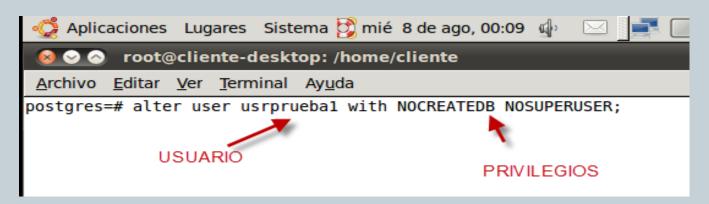
Crear un usuario llamado prueba con password prueba que tenga privilegios de superusuario que pueda crear bases de datos y que puedan crear roles.

CREATE USER PRUEBA WITH ENCRYPTED PASSWORD 'PRUEBA' SUPERUSER CREATEDB CREATEROLE;

Ejemplo

Para poder quitar los privilegio de usuario se realiza lo siguiente

ALTER USER [USUARIO] WITH NOSUPERUSER NOCREATEDB NO CREATEROLE;



• Para poder eliminar un usuario del DBMS se utiliza el comando DROP de la siguiente manera:

DROP USUARIO [NOMBRE DEL USUARIO];
DROP ROLE [NOMBRE DEL ROLE];

• Para poder modificar atributos de un usuario o un rol se utiliza la sentencia ALTER.

ALTER USER prueba RENAME prueba1;
ALTER ROLE "prueba" PASSWORD 'prueba';
ALTER USER "prueba" PASSWORD 'prueba';

• Para poder crear un objeto ya sea base de datos, tablas se utiliza el comando crear de la siguiente manera:

CREATE DATABASE [NOMBRE DE LA BASE DE DATOS];

• Para crear una base de datos con dueño se crea de la siguiente manera.

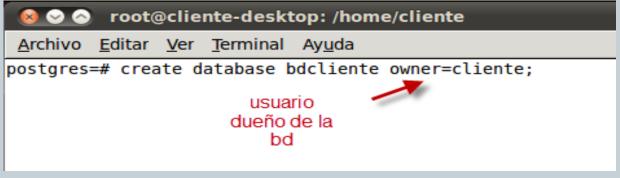
CREATE DATABASE [NOMBRE DE LA BASE DE DATOS] OWNER [NOMBRE_USUARIO];

• La sentencia para poder crear una table se lo realiza de la siguiente manera.

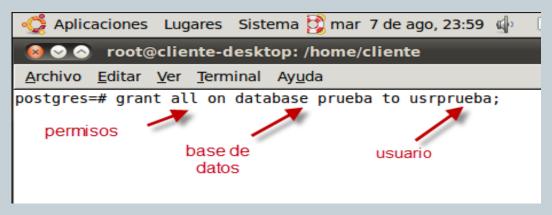
CREATE TABLE [NOMBRE TABLA](ATRIB1, ATRIB2...);

• Para crear una base de datos con dueño se crea de la siguiente

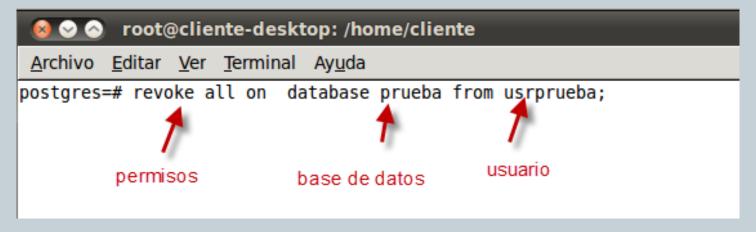
manera.

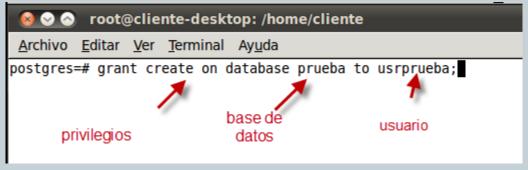


Comando para asignar permisos es GRANT



Comando para quitar permisos es REVOKE





• Creando tablas (clases) en PostgreSQL

PostgreSQL SQL soporta los tipos habituales de SQL como: integer, float, real, smallint, char(N), varchar(N), date, time, and timestamp, así como otros de tipo general.

Ejemplo.

```
create table DOCENTE(cod_doc int, nombre_doc varchar(30),apellido_pat varchar(30), apellido_mat varchar(30),profesion varchar(30));
```

• Para poder eliminar un objeto ya sea base de datos, tablas se utiliza el comando DROP de la siguiente manera:

DROP DATABASE [NOMBRE DE LA BASE DE DATOS];
DROP TABLE[NOMBRE DE LA TABLA];

 Para poder modificar datos de un objeto ya sea base de datos usuarios, tablas se utiliza la sentencia ALTER

ALTER DATABASE prueba RENAME prueba1; ALTER TABLE prueba RENAME prueba1;

POSTGRES: ADMINISTRACION

• Para poder eliminar un objeto ya sea base de datos, usuarios, tablas se utiliza el comando DROP de la siguiente manera:

```
DROP DATABASE [NOMBRE DE LA BASE DE DATOS];
DROP USUARIO[NOMBRE DEL USUARIO];
DROP TABLE[NOMBRE DE LA TABLA];
```

 Para poder modificar datos de un objeto ya sea base de datos usuarios, tablas se utiliza la sentencia ALTER

```
ALTER USER prueba RENAME prueba1;
ALTER ROLE "prueba" PASSWORD 'prueba';
ALTER USER "prueba" PASSWORD 'prueba';
ALTER DATABASE prueba RENAME prueba1;
ALTER TABLE prueba RENAME prueba1;
```