**Evidencia día 1 semana 13**

**Uso de aplicacion**



**Descripción**

En esta Clase se realizo el anlisis la plataforma de trabajo en postgres mara el trabajo con la bases de datos.

**Reflexión**

En mi analisis de profundisacion de los temas relacionados:

**VISTAS**

Las vistas son pseudo-tablas. Es decir, no son tablas reales; sin embargo aparecen como tablas ordinarias para SELECCIONAR. Una vista puede representar un subconjunto de una tabla real, seleccionando ciertas columnas o ciertas filas de una tabla ordinaria. Una vista puede incluso representar tablas unidas. Debido a que a las vistas se les asignan permisos separados, puede usarlos para restringir el acceso a la tabla de modo que los usuarios vean solo filas o columnas específicas de una tabla.

Una vista puede contener todas las filas de una tabla o filas seleccionadas de una o más tablas. Se puede crear una vista a partir de una o varias tablas, lo que depende de la consulta PostgreSQL escrita para crear una vista.

Las vistas, que son una especie de tablas virtuales, permiten a los usuarios hacer lo siguiente:

Estructurar los datos de una manera que los usuarios o las clases de usuarios encuentren natural o intuitiva.

Restrinja el acceso a los datos de modo que un usuario solo pueda ver datos limitados en lugar de una tabla completa.

Resuma datos de varias tablas, que se pueden utilizar para generar informes.

Dado que las vistas no son tablas ordinarias, es posible que no pueda ejecutar una instrucción DELETE, INSERT o UPDATE en una vista. Sin embargo, puede crear una REGLA para corregir este problema de usar DELETE, INSERT o UPDATE en una vista.

Creando Vistas

Las vistas de PostgreSQL se crean utilizando la declaración CREATE VIEW . Las vistas de PostgreSQL se pueden crear a partir de una sola tabla, varias tablas u otra vista.

La sintaxis básica de CREATE VIEW es la siguiente:

CREATE [TEMP | TEMPORARY] VIEW view\_name AS

SELECT column1, column2.....

FROM table\_name

WHERE [condition];

Puede incluir varias tablas en su declaración SELECT de una manera muy similar a como las usa en una consulta SELECT de PostgreSQL normal. Si la palabra clave opcional TEMP o TEMPORARY está presente, la vista se creará en el espacio temporal. Las vistas temporales se eliminan automáticamente al final de la sesión actual

**Interfaz Python**

PostgreSQL se puede integrar con Python usando el módulo psycopg2. sycopg2 es un adaptador de base de datos PostgreSQL para el lenguaje de programación Python. psycopg2 fue escrito con el objetivo de ser muy pequeño y rápido, y estable como una roca. No es necesario instalar este módulo por separado porque se envía, de forma predeterminada, junto con la versión 2.5.x de Python en adelante.

Si no lo tiene instalado en su máquina, puede usar el comando yum para instalarlo de la siguiente manera:

$yum install python-psycopg2

Para utilizar el módulo psycopg2, primero debe crear un objeto Connection que represente la base de datos y luego, opcionalmente, puede crear un objeto cursor que le ayudará a ejecutar todas las sentencias SQL.

API del módulo Python psycopg2

Las siguientes son rutinas importantes del módulo psycopg2, que pueden ser suficientes para trabajar con la base de datos PostgreSQL desde su programa Python. Si está buscando una aplicación más sofisticada, puede consultar la documentación oficial del módulo Python psycopg2.