Imagen que contiene alimentos

Descripción generada automáticamente

Universidad del Rosario

MACC

Base de datos

Claudia Hernández

Johan Caro

Andrés Yañez

Juanita Robles

Semestre 2, 2021

**GitHub:** <https://github.com/TheChieft/Proyecto_IgDatos>

**Ilustración 1 MODELO RELACIONAL 3RA FORMA NORMAL**

Interfaz de usuario gráfica, Texto, Aplicación, Escala de tiempo

Descripción generada automáticamente

**Descripción proceso de carga de información**

Fuente de datos:

https://www.datos.gov.co/Seguridad-y-Defensa/Reporte-Homicidios-y-Homicidios-accidente-de-tr-ns/ha6j-pa2r

<https://www.datos.gov.co/Mapas-Nacionales/Departamentos-y-municipios-de-Colombia/xdk5-pm3f>

El primer problema que tuvimos con la carga de datos fue que ambas fuentes de datos tenían el atributo Código DANE del municipio, pero tenían un problema el reporte de homicidios lo tenía en formato entero mientras que el de mapas nacionales lo tenía en formato decimal poniendo antes de la como el primer término (que corresponde al código departamental) y luego de la como el resto del código municipal. El grupo soluciono esto usando la función buscar y reemplazar de Excel para quitar la coma y que quedara como un numero entero (en los archivos .csv del GITHUB esto ya está realizado). Así que después de realizar esto las dos columnas de Código DANE del municipio ya tenían los datos en el mismo tipo de datos, un integer. Después de esto nos dimos cuenta de que había otro problema, la tabla Código DANE del municipio de los datos de Homicidios tenían todos tres ceros añadidos al final, así que decidimos revisar en la página del DANE cuál de los dos códigos era correcto y esta búsqueda nos fijó que el código correcto era el de la fuente de mapas nacionales entonces en él .csv de Homicidios volvimos a utilizar la función buscar y reemplazar de Excel para reemplazar “000,” por una “,”, así quitamos los tres ceros innecesarios al final y poniéndole la coma al final nos aseguramos de que solo quite tres ceros del final y no intermedios. Después de esto ya los datos estaban listos para ser cargados en la base de datos

Lo primero que se realizó en el SQL fue crear una data base (el código SQL de la creación estará en el repositorio de GITHUB) y luego agregarle dos tablas (datos\_no\_normalizado y tabla\_municipios\_departamentos en el archivo SQL) con los mismos atributos de las dos fuentes de datos para poder cargarlos en el PostgreSQL. Después se cargaron los datos con la función COPY de SQL a ambas tablas. Luego utilizando la función Upper de Postgres todos los datos de la tabla tabla\_municipios\_departamentos se pasaron a mayúscula para que ambas fuentes de datos tengan todos sus datos en mayúscula para que no haya problemas a la hora de hacer comparaciones entre datos.

Lo segundo que se realizo fue la creación de las tablas Departamento y Municipio, utilizando los datos de la fuente de datos #2, se crearon en dos tablas diferentes para respetar los datos atómicos de la primera formal ya que la tabla original tenía redundancia y repetición de los datos.

Después se realizó la creación de las tablas Tipo\_arma, Grupo\_etario, Tipo\_homicidio y Genero\_victimario, la creacion de estas cuatro tablas se realizó para optimizar la base de datos ya que los datos originales tomaban estos datos como atributos del tipo varchar, pero al separarlo con tablas que tienen como datos un ID que da un serial asociado a ese varchar, se logra que la tabla principal (Homicidios\_colombia) en vez de tener que cargar más de un millón de datos del tipo varchar, cargue un millón de datos pero del tipo integer haciendo que necesite menos recursos para cargarse.

Luego se creó la última tabla la cual es Homicidios\_Colombia, que recibe llaves foráneas de todas las tablas anteriores menos de la de departamentos para generar una tabla con los siguientes atributos:

* ID\_caso
* Fecha\_homicidio
* Numero\_victimas
* ID\_tipo\_arma
* ID\_genero\_victimario
* ID\_grupo\_etario\_victimario
* ID\_tipo\_homicidio
* Codigo\_Municipio\_Dane

Finalmente, luego de la creación de las 7 tablas se realizó el paso de las dos tablas iniciales que contenían los datos de la fuente a nuestra base de datos (todo el código de esto está en el archivo DDL.sql y DML.sql que se encuentra en el GITHUB), después de pasar todos los datos a nuestras tablas borramos las dos tablas con los datos fuentes para que no haya tablas con datos repetidos.

**Posibles análisis base de datos**

Desde la elección de la fuente de datos nuestro objetivo para analizar los datos ha sido claro.

1. Queremos analizar los datos recolectados para saber cuáles son los municipios con más homicidios en los últimos 10 años y los municipios con menor cantidad de homicidios en este mismo lapso
2. Conocer el porcentaje de victimarios de homicidios en Colombia que son menores de edad
3. Conocer los tipos de armas más utilizadas a la hora de cometer un homicidio en Colombia
4. Conocer el porcentaje de hombres y mujeres victimarios en Colombia en los últimos 10 años
5. Conocer el departamento con más homicidios en los últimos 10 años
6. Conocer los asesinatos que han sucedido en el país, separados por años