DATA WAREHOUSE

REGISTRO DE ANIMALES DE COMPAÑÍA

María Dolores Sesmero Pozo
Juan Ángel Piqueras López

Contenido

1. RESUMEN

Un Data Warehouse (almacén de datos), es el instrumento que las organizaciones encontraron para cohesionar los datos de todos los departamentos en un único lugar. Provee de una manera efectiva y organizada un conjunto de datos masivo con el objetivo de ayudar a la toma de decisiones informadas.
Este trabajo consiste en la creación de un Data Warehouse a partir de datos extraídos de OpenData Euskadi, en nuestro caso un registro de identificación de los animales de compañía. A partir de estos datos se realizará el proceso ETL para formar el Data Warehouse, y posteriormente analizar y extraer conclusiones de los datos del mismo.

2. INTRODUCCIÓN

La fuente de información que hemos usado para el desarrollo del Data WareHouse es "http://opendata.euskadi.eus", y en concreto el tema seleccionado ha sido "Registro de identificación de animales de compañía de la CAE (REGIA)"

En esta fuente de información aparecían archivos csv que contienen los datos de “Araba”, “Bizkaia”, “Gipuzkoa” y “fueraEuskadi”. El archivo que usaremos es el de “fueraEuskadi.csv”, que contiene datos como la fecha de nacimiento de los animales, fecha de alta e implantación de chip en los mismos, así como el identificador del chip, el tipo de animal, sexo, raza, y localización del animal.

El objetivo de este Data WareHouse es llevar un registro de los animales de compañía que tienen implantados un chip, que puede servir para saber que raza de animal es más común que tenga implantado un chip, en qué municipios y provincias hay más animales registrados o el número de animales registrados por año, entre otras muchas funcionalidades.

3. DESARROLLO

En este apartado se explica todo lo relativo al procesamiento de los datos obtenidos como su extracción, transformación y carga, y diseño del Data WareHouse. Además, se informa sobre las herramientas y tecnologías utilizadas para realizar estas tareas.

3.1. TECNOLOGÍAS USADAS

- **Python**: usado como lenguaje de programación para el tratamiento de los datos, lectura del csv, limpieza y transformación del mismo, así como para la creación de las tablas de la Base de Datos e inserción de los datos en las mismas.
- **SQLite**: usado como sistema gestor de base de datos.
- **Visual Paradigm**: usado para realizar el diseño del Data WareHouse.

- **LibreOffice Calc:** para la visualización de los datos en el formato de csv.

- **Matplotlib.pyplot:** usados para mostrar los resultados obtenidos del Data WareHouse en gráficas.

3.2. DISEÑO DEL DATA WAREHOUSE

Nuestro Data WareHouse usa un diseño en estrella, que consiste en una tabla de hechos rodeada por una serie de tablas de dimensiones.

Las tablas de dimensiones son las siguientes:

- **Dimension_Sexo:** contiene el sexo del animal (macho o hembra).

- **Dimension_Aptitud:** almacena las distintas aptitudes que poseen los animales (caza, pastor, guarda y defensa y lujo y compañía).

- **Dimension_Raza:** representa las distintas razas de los animales.

- **Dimension_FechaImplantacion:** representa la fecha en la que le implantan el chip al animal, aparte de esta fecha aparece la fecha de nacimiento que será usada para crear una de las medidas que consiste en calcular la diferencia de tiempo que pasa entre que el animal nace hasta que le implantan el chip.

- **Dimensión_Localizacion:** representa el municipio y provincia donde vive el animal.

La tabla de hechos es:

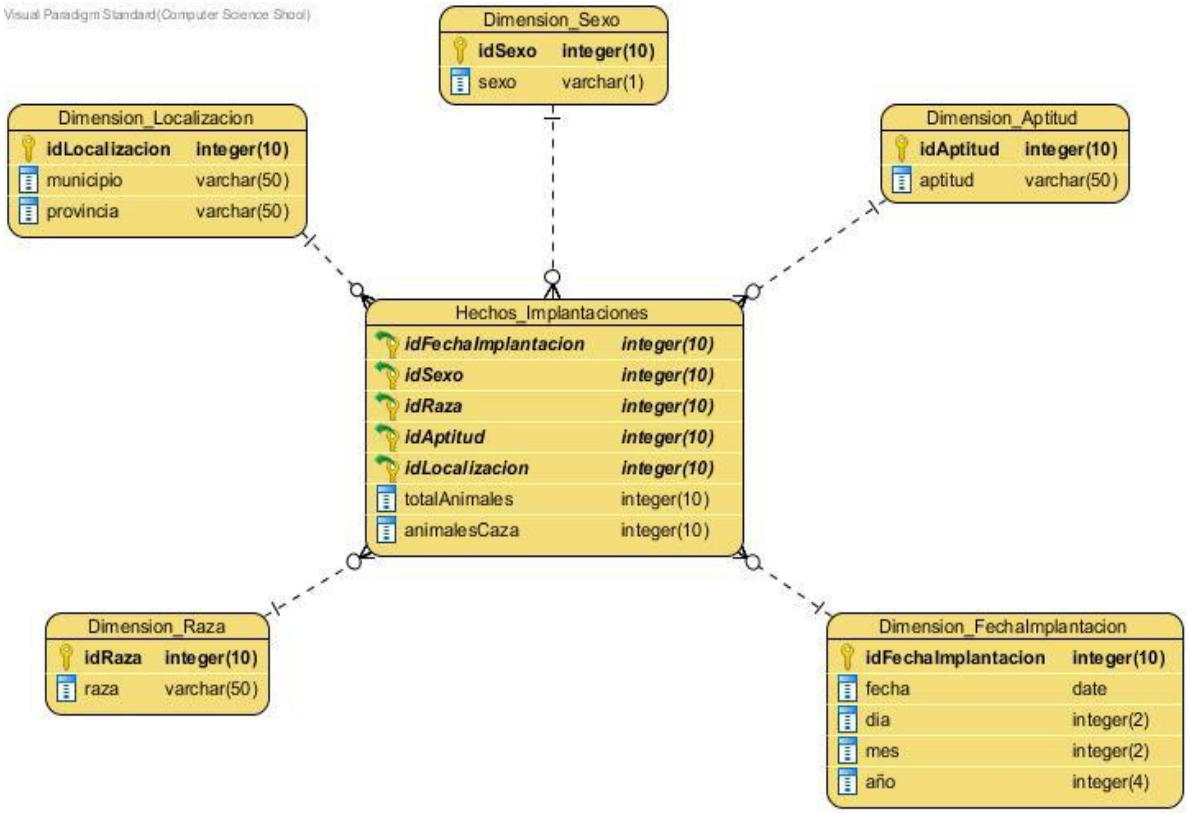
- **Hechos_Implantaciones**: la clave primaria de esta tabla está formada por las tablas ajenas de las tablas de dimensiones que relaciona. Las medidas que tiene la tabla de hechos son:

    - "difTiempoNacImp": representa la diferencia de tiempo (expresada en días) que existe entre el nacimiento de un animal y la implantación del chip.

    - "totalAnimales": representa el total de animales que existen con las especificaciones impuestas en las tablas de dimensiones.

    - "animalesCaza": representa el total de animales, de los registrados, que son de caza.

El diseño del Data WareHouse es el siguiente:



3.3. PROCESOS ETL

3.3.1. Extracción

Este proceso consiste básicamente en extraer los datos de las fuentes de información, ya sean internas o externas a la organización para posteriormente realizar el proceso de transformación y el de carga de los mismos. En nuestro caso los ficheros están en formato csv y fueron extraídos de OpenData Euskadi.

3.3.2. Transformación

Una vez que se tienen los datos en un formato uniforme, se debe realizar un preprocesado de los mismos, de tal forma que se eliminen las inconsistencias que puedan estar presentes a fin pasar los datos lo más limpios posibles al Data WareHouse. Para ello en esta etapa se procede a la eliminación de campos que eran irrelevantes para el desarrollo del almacén de datos, corrección de registros, comprobación de ausencia de algunos datos, etc.

En primer lugar, se realiza una visualización de algunos registros en busca de los tipos de incoherencias más comunes, como la presencia de valores vacíos, sin sentido (fechas erróneas) o errores en la escritura de los datos.

A continuación, procedimos con el filtrado de ciertos campos:

- Filtramos el campo “sexo” quedándonos solo con las filas donde aparecía en ese dato una “M” (macho) o “H” (hembra), ya que aparecían algunas filas con una “C” que no sabíamos a qué correspondía.

- En el campo “especie” filtramos por “Canino” o “Felino” ya que aparecían algunos datos raros, y poco comunes, como “Primate”.

- En el campo “provincia” eliminación de algunas traducciones de provincias al euskera.

- El campo “raza” no ha sido limpiado ya que contenía datos con múltiple variedad de formatos, algunas aparecían en español e inglés, otras también aparecían en euskera, algunas aparecían entre paréntesis con algún comentario respecto a la raza, etc, así que decidimos almacenar toda la información como un “string” y así no perder nada de información.

Después eliminamos los campos que no eran necesarios para nuestro Data WareHouse, quedándonos con los siguientes campos:

- numeroChip: usado para poder encontrar datos de animales duplicados ya que el único campo respecto a la información de un animal que no puede estar repetido.
- fecha_nacimiento y fecha_implantación del chip.
- sexo, raza y aptitud del animal: usado para distinguir los distintos tipos de animales de compañía almacenados.
- municipio y provincia del animal.

Por último, comprobamos la codificación con la que había que leer el csv ya que no reconocía bien ciertos caracteres como la “ñ” o las vocales con tilde.

Todas estas tareas son realizadas en un script llamado “cleanData.py” que lee el archivo fuente “fueraEuskadi.csv” y guarda todos los cambios en nuevo csv llamado “datos.csv” para trabajar con él a partir de ahora.

3.3.3. Carga

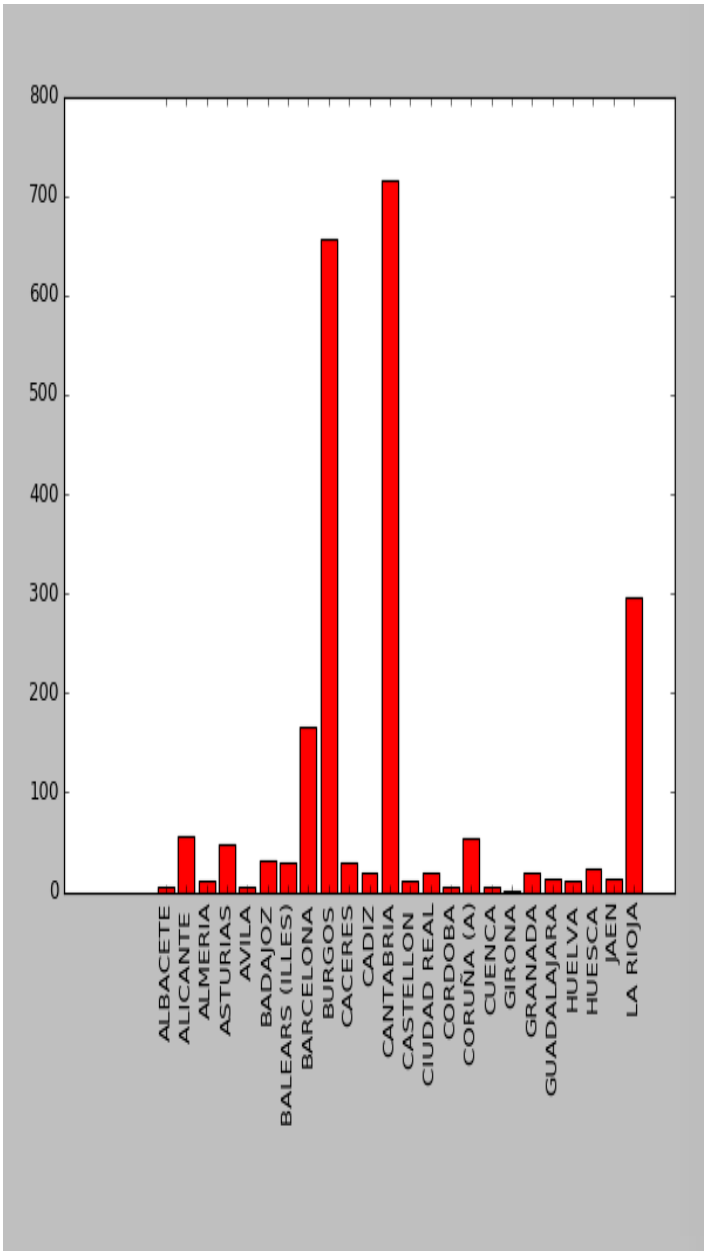
En esta parte del proceso, en primer lugar, se crea un script llamado “createTables.py” donde se crean las tablas necesarias siguiendo el diseño realizado del Data WareHouse. Para ello se tienen en cuenta los tipos de datos de los valores a introducir y la definición de las claves primarias necesarias. Además, se crean algunas tablas que usaremos como auxiliares para facilitar la inserción de los datos en las tablas de dimensiones y hechos correspondientes, que serán eliminadas después de su uso.

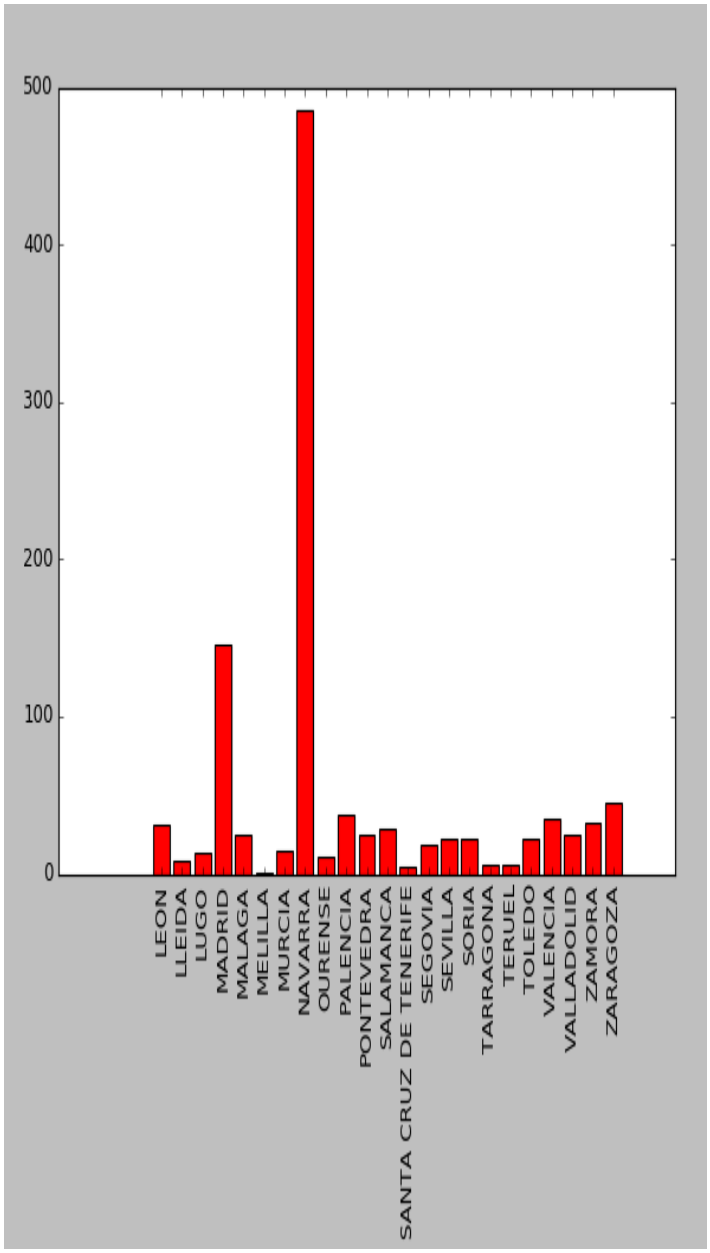
En segundo lugar, se crea otro script llamado “insertData.py” donde se realizan las funciones necesarias para la inserción de los datos en las tablas de dimensiones y de hechos.

4. RESULTADOS

Una vez creado el DataWarehouse, puede servir para llevar un registro de los animales que tienen implantado chip, así como realizar un análisis estadístico de cuáles son los animales que es más común que tengan chip. Realizamos una serie de consultas que pueden ser interesantes, en un script llamado "queries.py", las cuales son:

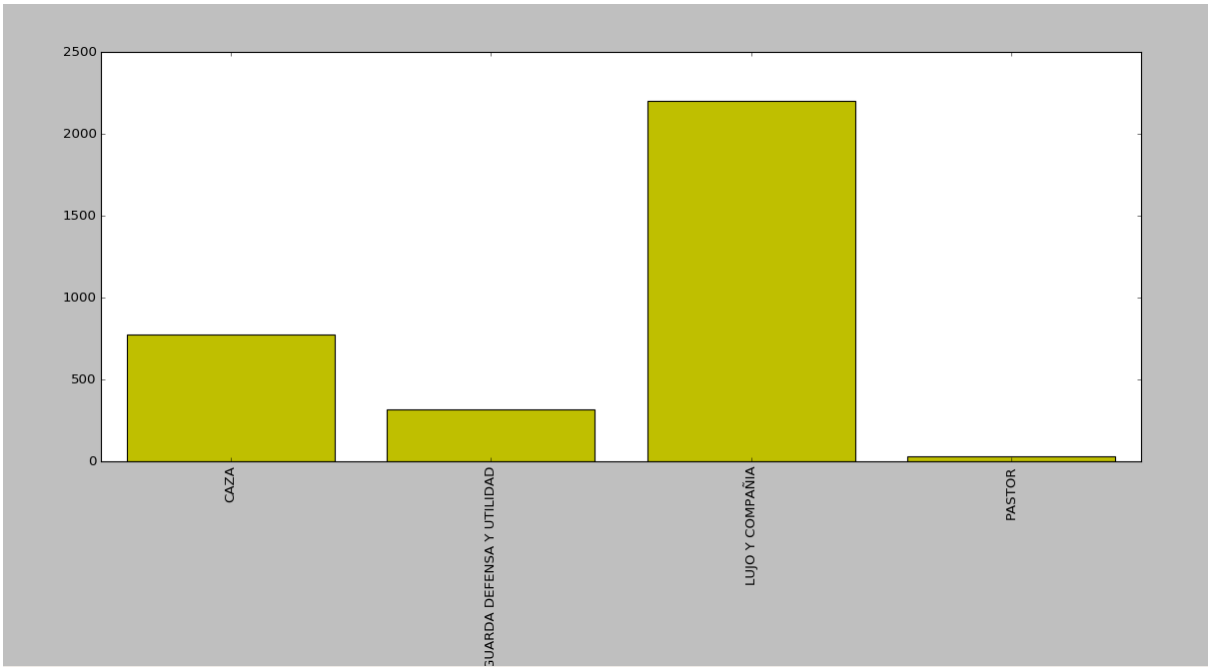
1. Número de animales que poseen chip por provincia.





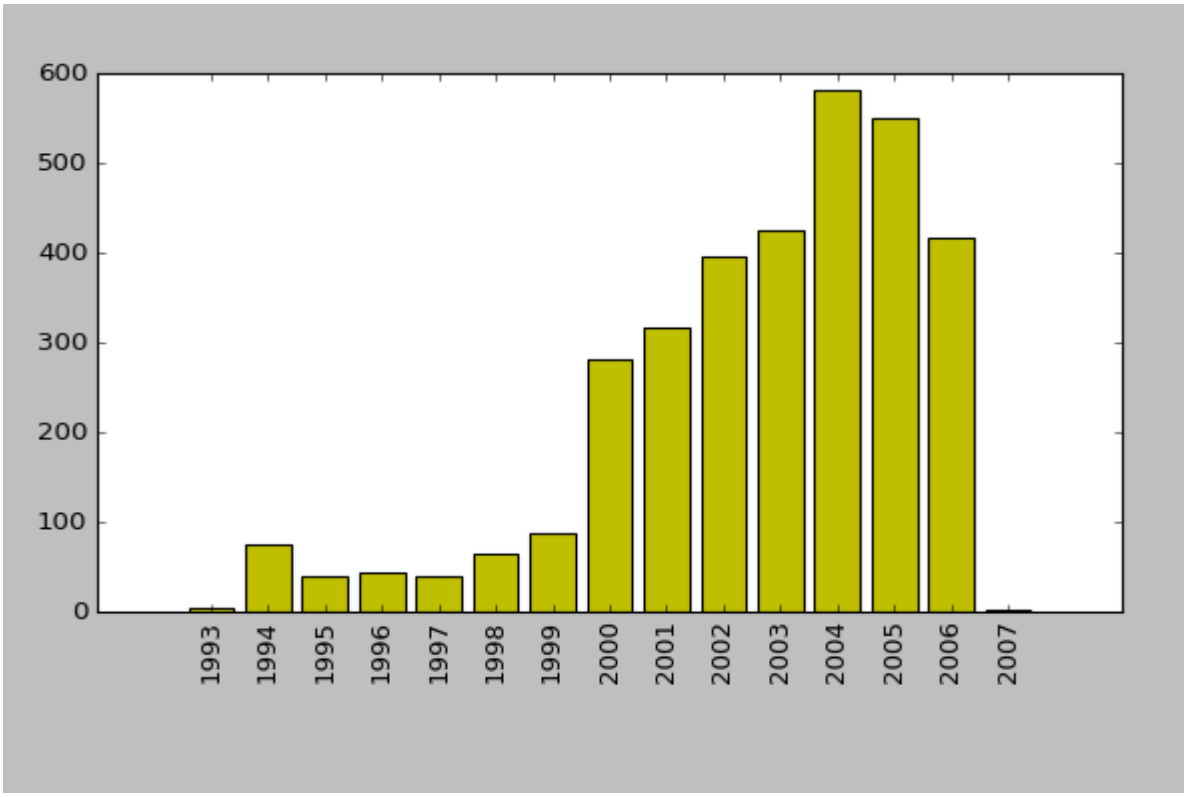
Se da la opción de introducir por teclado el nombre de la provincia que se desea consultar el número de animales registrados.

2. Número de animales que poseen chip por aptitud.



Se da la opción de elegir una de las cuatro aptitudes para mostrar solo por pantalla el número de animales registrados con dicha aptitud.

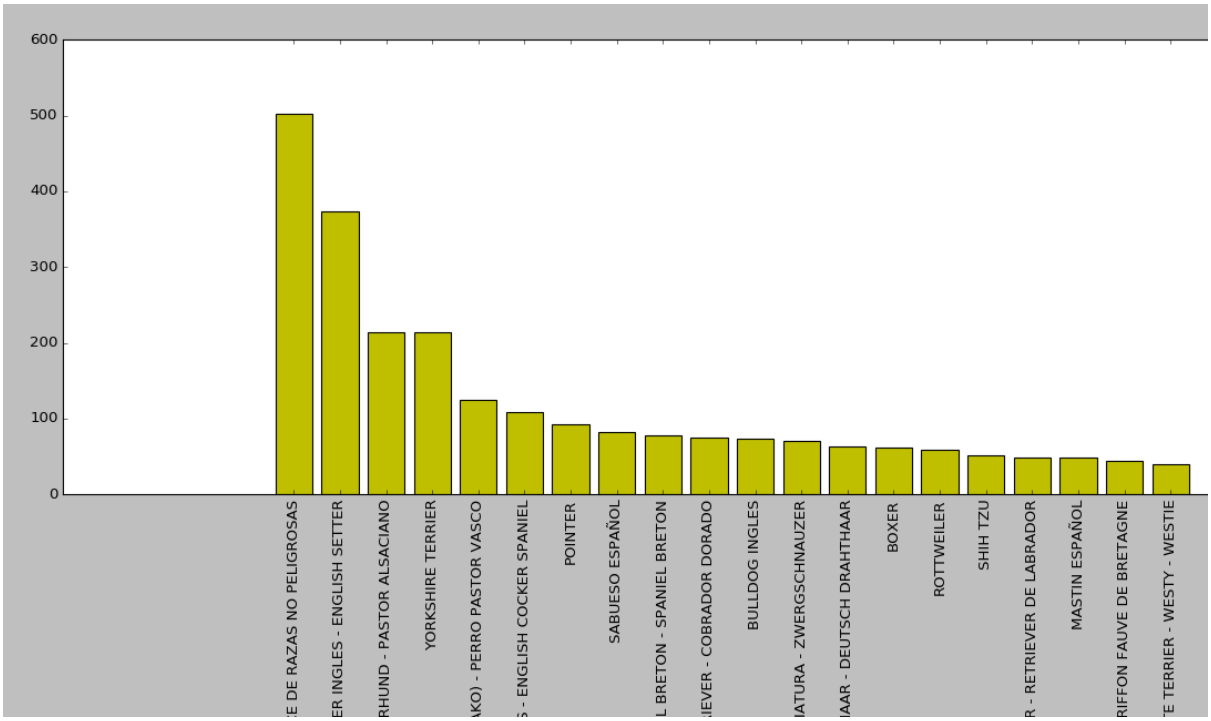
3. Número de animales por año de implantación.



Se permite la opción de introducir el año que se desea mostrar, e imprimiría por pantalla el número de animales registrados en ese año.

4. Número de animales por raza.

En este caso ya que existen un gran número de razas distintas y no se podía representar bien mostramos las 20 razas con mayor número de animales registrados.



5. CONCLUSIONES

El Data WareHouse ofrece un análisis rápido de resultado para la toma de decisiones, facilita el funcionamiento de las aplicaciones de los sistemas de apoyo a la decisión y proporciona un gran poder de procesamiento de la información, así como mayor flexibilidad y rapidez en el acceso a la información.

Además de almacenar los datos históricos, es de vital importancia el proceso ETL, para así poder almacenar algo útil y con valor. Es importante seleccionar buenos datos que se encuentren bien estructurados para que este Data Warehouse nos proporcione información relevante. De no ser así, el procesamiento de los datos seleccionados puede alcanzar complejidades que hagan que la implementación de dicho Data Warehouse no sea viable.

6. REFENRENCIAS

Data Warehouse – 00 - Conceptos Básicos
Data Warehouse – 01 - Análisis OLAP
Data Warehouse – 02 - Diseño en Estrella
Data Warehouse – 03 - Ejemplo de Diseño
Data Warehouse – 04 - Ejemplo ANuCla
Data Warehouse – 05 - Ejemplo IGDWEB
Data Warehouse – 06 - Ejemplo RPLE
Data Warehouse – 07 – Procesos ETL
Data Warehouse – 08 – Data Mining
Almacenes de Datos - Matilde Celma
SQLite Home Page
DB-API 2.0 interface for SQLite databases
matplotlib