

## **Estructura de datos.**

### **Código principal:**

#### Graficas.

Al ingresar a la opción 4 se despliega otro menú para que el usuario ingrese que datos desea graficar.

Para hacer las gráficas se extraen de los archivos de texto dos listas con los datos solicitados para hacer las gráficas.

Al finalizar la impresión de datos, se guardarán las imágenes, en el nombre como el tipo de gráfica, la fecha y hora, en la carpeta "Graficas".

#### Cálculos.

La opción 5 del menú principal son los cálculos, primero revisa si hay datos de lo contrario se regresa al menú, si hay datos continua con los cálculos de la moda, mediana y media con los datos de los archivos de texto, el peso.

Al igual que las gráficas, realiza dos listas con los archivos de texto y los cálculos se realizan con una función con los datos de la lista llamada 'flotantes'.

#### Conversión.

Realiza una conversión del id de los pokemon a 3 sistemas numéricos diferentes, para esto extrae una lista de los archivos de texto con los ids de los pokemon, después realiza otra lista 'id2' para organizar los datos.

## **Funciones**

Clean: Lo que realiza esta función, es la limpieza de pantalla. Extrae el sistema operativo del usuario mediante la función "os.name". y realiza la condicional, si es Unix(Posix), utilizara "Clear", pero si es otro, utilizara "cls".

#### Consulta.

En esta función se hace la consulta al api, datos es un diccionario en donde se extrae la información convertida de JSON a diccionario. A partir de este se extraen dos diccionarios 'pokemon' y 'generation', además de que se crea una lista donde se almacenara todos los que necesitamos para las opciones del menú.

Para extraer la primera parte de la información se necesita una segunda consulta que devuelve un diccionario, donde este diccionario contiene listas de diccionarios, en donde se almacena información que necesitamos, así que por medio de 'if' agregamos a nuestra lista lo que necesitamos y dejamos la demás información de lado.

La segunda parte de la información que viene en el diccionario que creamos 'generation', tiene una estructura de manera similar a la primera parte de la información, al igual que en la otra es necesaria una tercera llamada al api, que nos devuelve un diccionario que almacenamos en 'infoextra' y para extraer su información se utiliza el mismo método, comparar las keys y agregar la información necesaria a la lista 'elementos'.

Así al final, la lista se vacía en un archivo de texto nuevo y se imprime al usuario, y en caso de que ya exista un registro con el mismo id, no lo guardara. Para realizar el guardado de archivos, utilizamos 2 archivos de texto. Uno se nombrará con el "id", donde dentro de este, estará el nombre del archivo que tiene todos los datos, y después, se guardará otro archivo con el id, fecha y hora.

#### Impresión.

En esta función, se imprime todo el contenido de una lista seleccionada y además, irá guardando la información en un diccionario, separándolos mediante "Split" en la parte donde este un ":", para así, tener las claves y valores, si no existe, se sale. También le pregunta al usuario si quiere guardar la información mostrada en un archivo de Excel. En caso de que no exista la carpeta "Reportes", esta se creará y ahí estará el archivo "historial", con los datos, fecha y hora que se hizo la consulta.

#### Historial.

En esta función se necesita dos listas, donde en una se almacena las coincidencias de un patrón al nombre de los archivos de una carpeta y en otra el nombre del archivo. La cual se utilizará para la extracción de varias funciones.

Línea 22 y 24: En estas líneas, se extrae la hora local en decimales, para después convertirla al formato "[Y-M-D-H-M-S]" y utilizarla para el guardado de datos

Exc\_Int: Utiliza excepciones, para verificar que el tipo de dato no sea diferente a un número entero, y no pasara de ahí hasta que se cumpla.

#### Funciones2

Ext\_peso: En esta función, se solicita la lista con todos los archivos con id que hay en la carpeta, y también la dirección donde se localiza. Mediante el tamaño de la lista, se utiliza un for que va leyendo si existe el archivo, en caso de que exista, con la dirección y la variable "id" va a abrir el archivo donde está el nombre con fecha y hora, este se guardará en un string, y abrirá otro archivo que extraerá el peso y el nombre, el cual se guardará en una lista y las retornará.

Ext\_altura: En esta función, se solicita la lista con todos los archivos con id que hay en la carpeta, y también la dirección donde se localiza. Mediante el tamaño de la lista, se utiliza un for que va leyendo si existe el archivo, en caso de que exista, con la dirección y la variable

“id” va abrir el archivo donde está el nombre con fecha y hora, este se guardara en un string, y abrirá otro archivo que extraerá la altura y el nombre, el cual se guardara en una lista y las retornara.

Ext\_id: En esta función, se solicita la lista con todos los archivos con id que hay en la carpeta, y también la dirección donde se localiza. Mediante el tamaño de la lista, se utiliza un for que va leyendo si existe el archivo, en caso de que exista, con la dirección y la variable “id” va abrir el archivo donde está el nombre con fecha y hora, este se guardara en un string, y abrirá otro archivo que extraerá el id, el cual se guardara en una lista y lo retornara

Ext\_tipo: En esta función, se solicita la lista con todos los archivos con id que hay en la carpeta, y también la dirección donde se localiza. Mediante el tamaño de la lista, se utiliza un for que va leyendo si existe el archivo, en caso de que exista, con la dirección y la variable “id” va abrir el archivo donde está el nombre con fecha y hora, este se guardara en un string, y abrirá otro archivo que extraerá el tipo y lo ira agregando a una lista, después mediante la función “counter”, contara cuantas veces sale cada uno y lo retornara para la comparación de datos.

ImpExl: En esta función, se utiliza el diccionario creado en impresión. Primero se preguntará si el usuario quiere agregar los datos a un Excel. En caso de que si, intentara mediante excepciones abrir el archivo, si no existe lo crea, después se iterara el diccionario, y se ira agregando al Excel, las claves y valores para al final guardarlo. En caso de que exista, se leerá el número de filas que hay en el Excel ya creado y se utilizara el diccionario para agregar los valores. Se utilizó una excepción para verificar que el archivo de Excel está cerrado, si está cerrado, los datos se guardaran.