

# EVIDENCIA 2. PROYECTO INTEGRADOR

Ileana Tapia Castillo A01773374

Ilan Navarrete Morton A01773183

Máximo Rodríguez Cerecedo



### Planteamiento del problema:

Se plantea desarrollar un programa que sea capaz de modificar un archivo CSV en Python y que contenga las siguientes opciones:

- A. **Mostrar el detalle del repositorio.**
- B. **Buscar un registro de datos.**
- C. **Consulta un conjunto de registros.**
- D. **Modificar data existente.**
- E. **Ingresar un nuevo registro de datos.**
- F. **Remove un registro de datos existente.**
- G. **Finalizar.**

## OverView:

The screenshot shows a Python IDE with a file named `main.py`. The code defines a `menu()` function that prints a list of options and a table. The table has columns: `BRAND`, `PRODUCT`, `Brand code for Commercial`, and `http://lighttree.com/Commercial`. The table contains two rows: `Public Sector` and `Commercial`. The program also shows a distribution of spaces in a 6-field record: `[0, 0, 0, 0, 0, 0]`.

```
def menu():
    if op == "e" or op == "E" :
        print("Ingresar Datos del repositorio \n")
        sd ingresar_datos(data)
```

Problems OUTPUT DEBUG CONSOLE TERMINAL PORTS Python Debug Console

B. Buscar un registro de datos.

C. Consulta un conjunto de registros.

D. Modificar data existente.

E. Ingresar un nuevo registro de datos.

F. Remover un registro de datos existente.

G. Finalizar.8

Buscar Datos

Ingresa el numero del valor que tiene:

1) levelType

2) code

3) catalogType

4) name

5) description

-----

BRAND	PRODUCT	Brand code for Commercial	http://lighttree.com/Commercial
Public Sector			
Commercial			

-----

Contiene numeros ? : False| Contiene alfanumericos ? : False| Contiene alfabeticos ? : True| Contiene caracteres especiales ? : True

-----

La distribución de espacios vacios en el registro de 6 campos es [0, 0, 0, 0, 0, 0]

-----

Contiene 5 campos

Función finalizada

### documentación del código de un programa

---

#### Por cada módulo de implementación

**Modulo:** Mostrar datos

**Procedimiento Publico:** Una función que puede ser utilizada dentro de cualquier parte del programa solo con ser importada.

**Funcionamiento:** El módulo mostrar datos es una función la cual pide un argumento, en este caso archivo, que corresponde al archivo CSV con el cual vamos a trabajar. El método abre y lee el archivo guardando la información en la variable arch, posteriormente con ayuda de un bucle for iteramos cada registro de arch con ayuda del método readlines(), para guardar cada uno de ellos en una variable llamada lista, posteriormente apoyándonos de la variable espacios donde multiplicamos el numero de espacios por la longitud de la columna menos el largo del valor escrito en la lista, la variable ren nos ayudara a imprimir los elementos de la lista y a esta misma le sumamos los espacios, actualizamos estas 2 variables hasta conseguir todos los valores del registro e imprimirlos.

**Entrada:** Archivo CSV

**Salidas:** Tabla que muestra los datos del CSV de manera ordenada

**Modulo:** Mostar detalle

**Procedimiento Publico:** Una función que puede ser utilizada dentro de cualquier parte del programa solo con ser importada.

**Funcionamiento:** El modulo de ver detalle del repo es una función la cual pide como argumento el archivo CSV, esta función cuenta con 4 variables un contador de filas y un contador de columnas, una variable arch para abrir y leer la data contenida en el archivo CSV y por ultimo una variable lista para poder usarla para guardar los registros en listas temporales e iterar todos los elementos con ayuda de bucles for anidados, donde sumo el numero de columnas que se tienen y calculo la cantidad de campos que contiene el repositorio.

**Entrada:** Archivo CSV

**Salidas:** Cantidad de columnas y cantidad de registros que tiene el repositorio.

**Modulo:** Buscar datos

**Procedimiento Publico:** Una función que puede ser utilizada dentro de cualquier parte del programa solo con ser importada.

**Funcionamiento:** El módulo buscar datos es una función que recibe un parámetro, en este caso el archivo CSV, usamos la variable op para que el usuario ingrese un valor de cualquiera de las columnas del documento, luego el usuario proporciona el código correspondiente, y creamos la variable arreglo la cual es una lista vacía para introducir los datos , utiliza un bucle for donde realiza una iteración por cada uno de los registros utilizando el método readlines(), utilizamos la variable lista para guardar los datos, utilizamos un condicional if para comprobar si el código es igual a la posición de la lista agregarlo al arreglo y luego hacer un recorrido para comprobar el registro y validar si los datos contiene alfanuméricos, numéricos, alfabéticos o caracteres especiales.

**Entrada:** Archivo CSV

**Salidas:** Muestra la cantidad de registros, caracteres alfanuméricos y especiales que contiene además de los campos que se encuentran vacíos y el número de campos

**Modulo:** Consulta de datos

**Procedimiento Publico:** La función puede ser utilizada dentro de cualquier parte del programa solo con ser importada.

**Funcionamiento:** La función consulta de datos recibe el parámetro archivo, tiene una variable llamada op que le permite al usuario decidir qué valor tiene para poder consultar su registro, posteriormente le solicita el código de dicho registro para poder guardarlo en la variable code, similar a como se hizo anteriormente se genera la variable arreglo la cual es una lista vacía para introducir los datos , utiliza un bucle for donde realiza una iteración por cada uno de los registros utilizando el método readlines(), utilizamos la variable lista para guardar los datos, utilizamos un condicional if para comprobar si el código es igual a la posición de la lista agregarlo al arreglo, e igual imprime los campos y la distribución de espacios vacíos.

**Entrada:** Archivo CSV

**Salidas:** Los datos solicitados, la distribución y los campos vacíos.

**Modulo:** Ingresar Datos

**Procedimiento Publico:** La función puede ser utilizada dentro de cualquier parte del programa solo con ser importada.

**Funcionamiento:** La función recibe como parámetro el archivo CSV, después solicita los siguientes valores guardándolos en sus respectivas variables con el mismo nombre, LevelType, code, catalogType, name, description, sourceLink, crea una nueva variable llamada registro, donde concatena todos los datos recibidos, usamos arch para abrir el archivo y apoyándonos del método open con el parámetro “a”, escribimos el nuevo registro por último cerramos el archivo.

**Entrada:** Archivo CSV , LevelType, code, catalogType, name, description, sourceLink

**Salidas:** archivo con el nuevo registro

**Modulo:** Modificar Datos

**Procedimiento Publico:** La función puede ser utilizada dentro de cualquier parte del programa solo con ser importada.

**Funcionamiento:** La función recibe como parámetro el archivo CSV, después solicita el valor de alguna de las columnas, posteriormente el código de esa columna, luego creamos la variable arreglo la cual es una lista vacía para introducir los datos , utiliza un bucle for donde realiza una iteración por cada uno de los registros utilizando el método readlines(), utilizamos la variable lista para guardar los datos, utilizamos un condicional if para comprobar si el código es igual a la posición de la lista agregarlo al arreglo y luego hacer un recorrido para comprobar el registro y validar si existe o no. Después de desplegar el menú con los datos y su subíndice le daremos la opción al usuario de elegir que dato desea modificar, almacenando este dato en la variable fila y también tendrá que introducir el numero del campo que desea modificar, a la variable fila le restamos 1, debido a que nuestra lista comienza en 0, luego le solicitamos el valor con el que desea sustituir el dato y lo guardamos en modCampos, creo una nueva variable llamada filebuffer para guardar los datos de manera temporal y mediante un bucle for almacenamos los datos dentro de file buffer para después reescribirlos en el archivo

**Entrada:** Archivo CSV

**Salidas:** archivo CSV modificado

**Modulo:** Remover datos

**Procedimiento Publico:**

**Funcionamiento:** La función recibe como parámetro el archivo CSV, este archivo es guardado con ayuda del método open y con el parámetro “r” podemos leerlo y almacenarlo en la variable

arch, guardamos las líneas de este archivo en la variable registros con ayuda del método readlines(), creamos una nueva variable llamada count y la igualamos en 0, para que cuando realicemos la impresión de los registros podemos colocar el contador aun costado de cada registro y conocer su posición, esto le permitirá al usuario saber la posición del registro que desea eliminar, después se le pedirá indicar dicho valor que se almacenará en la variable fila, creamos de nuevo una variable filebuffer para almacenar los datos temporalmente. Mediante un bucle for agregaremos estos datos a el filebuffer omitiendo la fila a eliminar.

**Entrada:** Archivo CSV

**Salidas:** archivo CSV modificado con una fila eliminada

**Modulo:** Main

### Procedimiento Publico:

**Funcionamiento:** El módulo Main es la función que inicializara el programa, *importamos las librerías que creamos anteriormente para cada una de las opciones del programa, el separarlas en funciones y usarlas como métodos nos ayuda a mantener un orden en el código, además de asignarles nombres más cortos, posteriormente importamos al catalogo y lo guardamos en la variable data para usarla cada vez que ocupemos el código, mediante otra función llamada menú creamos una variable local llamada op para que el usuario elija la opción que desea ejecutar sobre su archivo CSV, utilizamos condicionales if para cada una de las funciones vistas anteriormente. Por medio de la función main mientras el menú no retorne un valor booleano false, el programa continuara con su ejecución*

**Entrada:** Archivo CSV

**Salidas:** Archivo CSV modificado en caso de hacer cambios y las opciones por consola dependiendo la elección del usuario.

### Reflexión Personal

**Ileana Tapia Castillo**

El realizar este proyecto fue para mí una experiencia retadora, debido a las restricciones del uso de funciones y métodos estandarizados en Python para lograr resultados similares. En este

proyecto tuve que crear mis propias funciones para poder realizar el paseo de datos, y el análisis de estos mismos, así como actualizar un archivo CSV sin ocupar una librería especializada. Al momento de realizar el proyecto el primer paso fue crear un diagrama de flujo para plantearnos su funcionamiento y las opciones que este tendría, apoyándome de los temas vistos con anterioridad usamos conceptos de crear librerías para poder separar nuestro código y hacerlo más ordenado, usando el conocimiento de funciones y valores booleanos y variables de tipo string, tuve crear un menú interactivo, que no cierra el programa hasta que el usuario haya terminado de realizar las acciones necesarias.

El trabajar con consultas de datos fue interesante ya que tenía que convertir el archivo CSV a una lista que después sería pasada a un arreglo para realizar iteraciones recolectando todos los datos de nuestro archivo, mostrarlos por pantalla, algunas funciones similares se podían ver en detalle y mostrar datos. Sin duda la parte más complicada para mí fue modificar datos preexistentes además de eliminar columnas específicas, en este tuve que hacer uso de un buffer, el cual es un espacio en memoria en el que se almacenan datos de manera temporal, luego convertirlo de nuevo a un csv, reescribiendo los datos y guardando estos cambios.

Ilan Navarrete Morton:

El aprendizaje obtenido con este proyecto va muy enfocado en el ámbito de la programación, sin embargo, también incluye el aspecto de resolución de problemas tratando de abordarlo desde diferentes perspectivas, ya que con el lenguaje Python hay muchas maneras de resolver el problema. Esta evidencia me ayudo y me hizo trabajar mi pensamiento lógico además de la investigación y el autoestudio, ya que a pesar de que en la clase se vieron temas con material. útil para este reto, había momentos en donde, para mí, me era de utilidad apoyarme de otros contenidos. Ya con las herramientas necesarias para empezar, me di cuenta de que también influía un factor muy importante para realizar este trabajo, y ese es la creatividad. Que a pesar de que la programación suele ser un proceso muy lógico y estructural es necesario un pensamiento flexible y que de con muchas formas de resolver problemas. Por último, aprendí de lo importante que es trabajar en equipo, además de darme cuenta que aún me queda mucho que aprender sobre este y otros lenguajes de programación.

Máximo Rodríguez:

El aprendizaje obtenido con este proyecto fue basto en cuestión de código, sin embargo, también llevo un aprendizaje en cuestión de trabajo en equipo y resolución de problemas en el proyecto, a los cuales les dimos solución mediante nuestro código, el cual, cumple con los requisitos solicitados exitosamente.

En resumen, este proyecto presenta un código sólido para gestionar catálogos de productos, con una interfaz que se puede adaptar fácilmente a otras bases de datos. Aunque existen áreas que podrían ser optimizadas, ya cumple de buena manera con las necesidades básicas e instrucciones solicitadas en el proyecto.