

PROYECTO GRUPAL

El Administrador de un conocido Bar de Valparaíso, tiene implementada una Base de Datos que permite registrar el Inventario de Productos que son utilizados como ingredientes para las preparaciones de tragos y bebidas (por ejemplo: Ruso Negro, Caipiriña, Tequila, Margarita sin Alcohol, etc.) o como productos que son vendidos en unidades (por ejemplo: Botella de Coca Cola, Botella de Ron, Botella de Pisco, etc.). La base de datos está compuesta por 3 archivos, en donde cada uno tendrá la siguiente estructura y valores (estos archivos quedarán publicados en el Aula Virtual):

PRODUCTOS.DAT						
IdProducto Texto(7)	Nombre Texto(50)	IdMarca Numérico(1)	IdTipoProducto Texto(7)	StockReal Numérico(3)	StockCritico Numérico(3)	Activo Texto(2)
PROD-01	RON	9	TIPO-1	10	7	SI
PROD-02	PISCO	4	TIPO-1	4	2	SI
PROD-03	PISCO SOUR	4	TIPO-1	13	10	SI
PROD-04	COCA COLA	1	TIPO-2	12	5	NO
PROD-05	BILS	7	TIPO-2	15	5	SI
PROD-06	PEPSI	3	TIPO-1	26	10	SI
PROD-08	PULPA FRUTILLA	5	TIPO-3	22	10	NO
PROD-11	LIMÓN	5	TIPO-3	12	2	SI
PROD-12	MARTINI BIANCO	2	TIPO-1	21	10	SI
PROD-13	AZÚCAR	5	TIPO-9	23	10	SI
PROD-14	MARTINI ROSSO	2	TIPO-1	7	5	SI
PROD-15	HIELO CUBOS	5	TIPO-6	3	5	SI
PROD-16	ROJO SANGRE	9	TIPO-8	1	5	SI
PROD-17	MENTA	5	TIPO-9	2	10	SI

MARCAS.TXT		
IdMarca Numérico(1)	Nombre Texto[20]	Activo Texto[2]
2	MARTINI	SI
5	EL HUERTO DE MAMÁ	SI
7	CCU	NO
9	MITJAMS	SI
1	COCA COLA COMPANY	SI
3	PEPSICO	NO
4	MISTRAL	SI

TIPOS_PRODUCTOS.DAT		
IdTipoProducto Texto[7]	Nombre Texto[20]	Activo Texto[2]
TIPO-1	LICORES	SI
TIPO-2	GASEOSAS	SI
TIPO-3	FRUTAS	SI
TIPO-5	DECORACION	NO
TIPO-6	HIELO	SI
TIPO-8	COLORANTES	SI
TIPO-9	SABORIZANTES	SI

ALGUNAS CONSIDERACIONES:

1. Los archivos PRODUCTOS.DAT y TIPOS_PRODUCTOS.DAT son archivos de acceso relativos, en cambio MARCAS.TXT es un archivo secuencial.
2. En el archivo PRODUCTOS.DAT, el campo StockReal señala la cantidad del producto que hay en existencias y el campo StockCritico señala cuánto es el mínimo que se debe contar del producto.
3. En el archivo TIPOS_PRODUCTOS.DAT hay registros en blanco, debido a que no hay un código de tipo de producto cuyo Hashing de en dicha posición (más adelante se explica lo que es un Hashing), lo mismo sucede en PRODUCTOS.DAT.
4. El largo del registro del archivo PRODUCTOS.DAT es de 73 caracteres.
5. El largo del registro del archivo TIPOS_PRODUCTOS.DAT es de 29 caracteres.
6. El área de overflow en el archivo PRODUCTOS.DAT comienza en el registro 101.
7. El área de overflow en el archivo TIPOS_PRODUCTOS.DAT comienza en el registro 21.
8. Una función Hashing es una operación aritmética que se realiza con un valor clave (identificador) de un registro, de tal forma que el resultado de la operación corresponde a una posición de registro dentro del archivo. Esta operación se realiza sobre archivos de datos cuya organización es relativa.
9. Si el Hashing da como resultado una posición dentro del área de overflow, se considera que los datos relacionados se encuentran en el área de overflow (recuerde que en esa área se recorren los registro secuencialmente).
10. Algunos ejemplos en donde el Hashing puede dar la misma posición en el archivo PRODUCTOS.DAT: PROD-03, PROD-3, P-3, etc. (En todos estos casos el Hashing da por resultado 3).
11. En los tres archivos, el campo Activo señala si el producto, marca o tipo de producto, según sea el archivo, está eliminado lógicamente o no. Un registro marcado como inactivo (valor del campo Activo en "NO") es un registro que se encuentre eliminado lógicamente (el registro permanece en el archivo, pero no puede ser utilizado). Un registro queda como inactivo cuando al querer eliminarlo físicamente esto no es posible debido a que un registro en otro archivo hace referencia a él. Por ejemplo, se desea eliminar la marca "MARTINI" en el archivo MARCAS.TXT, pero no es posible, ya que existe un producto en el archivo de PRODUCTOS.DAT que hace referencia a esta marca, por lo tanto, se asigna al campo Activo el valor "NO".

SE PIDE Implementar un programa en Python que permita realizar los siguientes procesos, los que deberán ser presentados en un menú:

1. **Agregar Producto.** Este proceso deberá solicitar el ingreso por teclado de Id del producto por agregar. A continuación, se deberá verificar si el Id ya existe en el archivo PRODUCTOS.DAT, para ello deberá aplicar la función Hashing, la que corresponde a los 2 dígitos de la derecha del Id del



producto (pueden ser 1 o 2 dígitos), para dar con su posición dentro del archivo. A continuación, se deberá verificar alguno de estos escenarios:

- a. **El registro está vacío.** Se pide el ingreso por teclado del Nombre, Id de la Marca del producto, Id del Tipo de producto, Stock Real y Stock Crítico. Finalmente, se guardan los datos como un registro en dicha posición.
- b. **El registro está ocupado por otro producto.** Se deberá saltar al área de overflow para buscar el Id del producto, recorriendo los siguientes registros de esta zona secuencialmente. Si existe, se deberá mostrar un mensaje de error. Si no existe, se pide el ingreso por teclado del Nombre, Id de la Marca del producto, Id del Tipo de producto, Stock Real y Stock Crítico. Finalmente, se guardan los datos como un registro al final del archivo.
- c. **El registro está ocupado por el mismo producto.** Se deberá mostrar un mensaje de error indicando el Id del producto ya existe.

Antes de guardar los datos, en la posición respectiva, se tendrá que validar que el Id de la Marca exista en el archivo MARCAS.TXT y el Id del tipo de producto en el archivo TIPOS_PRODUCTOS.DAT. Si no existe en algunos de ellos, se deberá pedir los datos restantes para ser añadidos en los respectivos archivos. Finalmente, se almacenan los datos en el archivo PRODUCTOS.DAT guardando el valor "SI" en el campo activo.

Para este proceso, se deberá realizar un acceso directo a los archivos PRODUCTOS.DAT y TIPOS_PRODUCTOS.DAT y un acceso secuencial en el archivo MARCAS.TXT

2. **Modificar Marca.** Este proceso deberá solicitar el ingreso por teclado del Id de la marca. Este valor debe existir en el archivo MARCAS.TXT. Si no existe, se deberá mostrar un mensaje de error. Si existe, se deberá solicitar el ingreso por teclado del nuevo Nombre de la marca. Finalmente, se almacenan los datos en el archivo MARCAS.TXT guardando el valor "SI" al campo activo. Recuerde que no se puede modificar directamente los datos de un registros, dado que el archivo se puede abrir de lectura o escritura, pero no de ambas formas.

El acceso a los registros en el archivo MARCAS.TXT debe ser secuencial.

3. **Eliminar Tipo de Producto:** Este proceso deberá solicitar el ingreso por teclado del Id del tipo de producto por eliminar. A continuación, se deberá aplicar la función Hashing, obteniendo los 2 últimos dígitos del Id del tipo de producto (puede ser que sean 1 o 2 dígitos). Luego, se deberá verificar si, en la posición obtenida al aplicar la función Hashing, el registro está ocupado o no, dando 3 escenarios:
 - a. **El registro está vacío.** Se deberá mostrar un mensaje de error, indicando que el Id del tipo de producto no existe.
 - b. **El registro está ocupado por otro tipo de producto.** Se deberá saltar al área de overflow para buscar el Id del tipo de producto, recorriendo los siguientes registros de esta zona secuencialmente. Si no existe, se deberá mostrar un mensaje de error. Si existe, se deberá verificar si existe algún producto en el archivo PRODUCTOS.DAT que haga referencia al tipo de producto por eliminar (se deberá hacer una búsqueda secuencia en el archivo PRODUCTOS.DAT). De existir, se asigna el valor "NO" al campo Activo y se guardan los



datos (se reescribe el registro) en el archivo TIPOS_PRODUCTOS.DAT. De no existir, se reescribe el registro asignando espacios en blanco a todos los campos.

- c. **El registro está ocupado por el mismo producto.** Se deberá verificar si existe algún producto en el archivo PRODUCTOS.DAT que haga referencia al tipo de producto por eliminar (se deberá hacer una búsqueda secuencia en el archivo PRODUCTOS.DAT). De existir, se asigna el valor "NO" al campo Activo y se guardan los datos (se reescribe el registro) en el archivo TIPOS_PRODUCTOS.DAT. De no existir, se reescribe el registro asignando espacios en blanco a todos los campos.

OBSERVACIONES:

- Todas las operaciones se deben realizar sobre el o los archivos respectivos. No se puede hacer uso de arreglos ni alterar la estructura de los archivos.
- Es muy importante que respete los nombres de los archivos de datos y de la estructura que estos tienen. La revisión se realizará empleando archivos con datos distintos y que son de pertenencia del profesor.
- Al principio, si los archivos de datos no existen deberán ser creados.

ENTREGA:

- Este trabajo se debe realizar en grupo de entre 3 y 5 estudiantes, ni más ni menos. Si no cumple esta condición tendrá 5 puntos de descuento.
- Cada grupo designará a uno de los integrantes como jefe de grupo, quien será responsable de la entrega de la tarea.
- En las primeras líneas de su programa debe anotar los nombres de cada integrante del grupo, comenzado por el jefe de grupo. Cada nombre debe anotarse en una línea por separado.
- El profesor podrá someter a los estudiantes a una interrogación para verificar que han participado en el desarrollo del programa y que tienen conocimiento del código escrito.
- Se dispondrá de un enlace en el AULA VIRTUAL dentro de la sección TAREAS. En este enlace el jefe de grupo deberá subir **solamente** el archivo .py el cual deberá llevar por nombre de archivo el nombre y apellido del jefe de grupo, por ejemplo: JUAN SOTO.py
- Cualquier archivo adicional o en otro formato que se suba no será considerado.
- Si no se cumple con estas instrucciones, la nota sufrirá 5 puntos de descuento importante.
- **Dispone hasta el domingo 26 de noviembre, hasta las 23:59, para hacer entrega del trabajo.**