Instrucciones Miniproyecto 1

Descripción

En este miniproyecto deberás completar un código, utilizando estructuras de

datos adecuadas, que luego se utilizará para realizar una serie de consultas a

una base de datos de películas y así obtener la información solicitada.

Objetivo

El objetivo de este miniproyecto es aplicar los conocimientos de tipos de datos,

ciclos, condicionales y estructuras de datos secuenciales para procesar y

guardar la información entregada. Una vez hecho esto, se deberá completar

diversas consultas que serán las que se utilizarán para obtener la información

buscada.

Archivos entregados

Para este trabajo, se entregará una carpeta MP1.zip que contiene dos archivos:

● main.py: Corresponde al archivo principal del programa, en el que deberás

cargar la información entregada y completar las consultas que se solicitan.

Este es el único archivo que debes modificar.

● peliculas.csv: Este archivo contiene información sobre diversas películas.

Cada línea contiene los datos de una película separados por comas (“,”),

de la siguiente forma:

titulo,popularidad,voto\_promedio,cantidad\_votos,generos

Donde los significados de cada atributo son los siguientes:

o titulo: Es el nombre de la película.

o popularidad: Número que indica la popularidad de la película.

Entre más alto, más popular es actualmente la película.

o voto\_promedio: Es un número que indica el puntaje promedio de

la película.

o cantidad\_votos: Es un número que indica la cantidad de votos que

ha recibido la película.

o generos: Representa un listado de géneros a los que pertenece

una película, separados por un punto y coma (“;”). Por ejemplo, si

en la columna “generos” de una película se encuentra el texto

“Adventure;Action;Science Fiction”, significa que la película es

categorizada bajo esos tres géneros.

Trabajo a realizar

El código que se entrega contiene implementado el flujo del menú principal, el

cual recibe un input indicando la consulta a realizar, abre el archivo de películas

y almacena cada línea de este en una lista y en base a esto ejecuta la función

correspondiente para finalmente mostrar el resultado en consola. Para asegurar

su funcionamiento, debes realizar las siguientes tareas:

Parte 1: Cargar los datos.

La primera parte del trabajo consiste en guardar los datos del archivo de

películas en diversas estructuras. Como se indicó anteriormente, el código

entregado ya se encarga de abrir el archivo de películas y generar una lista en

donde cada elemento es una línea del archivo, por lo que debes trabajar en base

a esta lista.

Para esto, debes completar la función cargar\_datos(lineas\_archivo) que se

encuentra en el archivo main.py, la cual tiene como argumento la lista de

películas extraídas del archivo, es decir, cada línea del archivo en formato string,

y en donde debes crear las siguientes estructuras:

• generos\_peliculas: Es una lista que debe almacenar todos los géneros

distintos de películas que existen en el archivo. Como una película puede

tener más de un género, debes ingresar los géneros de cada película por

separado a la lista. Además, como las películas pueden compartir

géneros, debes verificar que cada género sea guardado en la lista una

sola vez.

• peliculas\_por\_genero: Es una lista de tuplas con el formato (genero,

[peliculas]), en donde el primer elemento corresponde al nombre de un

género de película, y el segundo elemento es una lista de películas que

poseen dicho género. Por cada género distinto de película, debería haber

exactamente una tupla. Sin embargo, como una película puede tener más

de un género, puede ocurrir que una película este en más de una tupla.

• info\_peliculas: Es una lista de tuplas con el formato

(titulo,popularidad,voto\_promedio,cantidad\_votos,[generos]), en donde el

último elemento debe ser una lista de los géneros de la película. Por cada

película, debería haber exactamente una tupla que la represente.

Finalmente, una vez que creadas las estructuras, debes hacer que la función las

retorne como una tupla, en donde los elementos deben poseer el siguiente

orden: generos\_peliculas, peliculas\_por\_genero, info\_peliculas.

Parte 2: Completar las consultas.

La segunda parte del trabajo consiste en completar las consultas que se

utilizarán para obtener información sobre las películas. Para esto, debes

completar las siguientes funciones que se encuentran en el archivo main.py:

• obtener\_puntaje\_y\_votos(nombre\_pelicula): Esta función recibe un

string, que corresponde al nombre de una película y debe retornar una

tupla, donde el primer elemento debe ser el puntaje promedio de la

película y el segundo elemento debe ser la cantidad de votos que tiene.

• filtrar\_y\_ordenar(genero\_pelicula): Esta función recibe un string que

corresponde a un género de película. Debes filtrar todas las películas que

sean de este género y retornar sus nombres en una lista. Además, esta

lista debe estar ordenada según orden alfabético inverso (es decir,

películas con z primero y al final películas con a).

• obtener\_estadisticas(genero\_pelicula, criterio): Esta función recibe

un string que corresponde a un género de película. Debes analizar las

películas que sean de este género y obtener el máximo, mínimo y

promedio de los valores dados por criterio (que puede ser “popularidad”,

“voto promedio”, “cantidad votos”). Esto debe ser entregado en formato de

lista con el formato [max, min, promedio], en el orden mostrado.

Para procesar la información requerida en las funciones, debes utilizar las

estructuras de datos creadas en el apartado anterior que consideres pertinentes.

No es necesario que las utilices todas, pero sí debes usar al menos una.

Importante: Método split()

Para que puedas realizar tu evaluación, deberás usar un método para strings de

Python llamado split(), el cual permite separar elementos de un string y

almacenarlos en una lista. Este tiene el siguiente formato:

lista = variable\_string.split(separador)

En donde variable\_string será una variable que contenga un string cualquiera,

y separador corresponde a otro string, el cual Python buscará para hacer las

separaciones del string en distintos elementos de una lista.

A modo de ejemplo, si tengo el siguiente código:

variable = “Amarillo-Rojo-Azul”

lista = variable.split(“-”)

print(lista)

El resultado del print debería ser ['Amarillo', 'Rojo', 'Azul'], efectivamente

separando los tres elementos del string según el separador dado.

Archivos a entregar

Debes entregar el archivo main.py con las definiciones de funciones que hacen

funcionar tu programa.

Consideraciones generales

• La evaluación deberá realizarse utilizando únicamente los contenidos del

curso vistos hasta la semana a la cual corresponde el miniproyecto

(semana 2). Es posible utilizar contenidos vistos tanto en las clases como

ayudantías. El no seguir esta consideración implicará una sanción en la

nota de la evaluación

• Se encuentra estrictamente prohibido el uso de herramientas de

generación de código de cualquier tipo. Si se hace uso de algún recurso

externo (libro, página web, entre otros), este debe ser citado

apropiadamente en el código. El no seguir estas consideraciones

implicará la calificación mínima de la evaluación (1.0)

• En el archivo main.py viene implementado un menú en consola que

pueden utilizar para probar el funcionamiento de su código. Para esto,

basta ejecutar el archivo e ir ingresando en consola la información

solicitada, con lo que se mostrará en consola el resultado de la consulta.

Un detalle importante para considerar es que el código está hecho para

recibir exactamente lo que se espera, es decir, deben tener cuidado de

correrlo sólo cuando ya hayan completado las definiciones de funciones

que se solicitan e ingresar inputs válidos, ya que en caso contrario el

código no funcionará. Pueden asumir que siempre se les preguntará por

información existente y correcta.

• La corrección de este miniproyecto se realizará mediante la ejecución de

su programa y la realización de consultas, por lo que debes asegurarte de

que el programa funcione y entregue en consola la información buscada.

• En este caso, todo el flujo del programa principal viene implementado,

pero es recomendable prestar atención a su funcionamiento ya que en

futuros proyectos esto deberá ser implementado por ustedes.

• El miniproyecto entregado con una extensión que no corresponda a lo

solicitado en las instrucciones o que no se pueda ejecutar, será calificado

con la nota mínima (1.0), sin derecho a una nueva entrega