Reto 5: Grupo P66

 Número máximo de ficheros: 1  
Tipo de trabajo:  Individual

Análisis Base de Datos de Libros

Los amantes de la lectura encuentran gran cantidad de portales en internet en donde es posible calificar contenidos literarios tanto a nivel profesional como aficionado. Por solo nombrar algunos de estos sitios, tenemos:

- Goodreads ([https://www.goodreads.com/](https://www.imdb.com/))

-  Book Browse ([https://www.bookbrowse.com/](https://www.rottentomatoes.com/))

- Book Tib (https://booktrib.com/)

- The Story Graph (https://www.thestorygraph.com/)

Para procesar estadísticas asociadas a las libros, se suelen crear bases de datos en donde figuran los títulos y algunas de sus características a través de formatos de intercambio que pueden ser accedidas en línea (en algunos casos de manera gratuita).

Para este reto, vamos a tomar como referencia una recopilación de libros disponible en un proyecto público de Git Hub ([https://github.com/benoitvallon/100-best-books.git](https://github.com/erik-sytnyk/movies-list.git)). Los libros acá incluidos presentan la siguiente estructura (https://raw.githubusercontent.com/benoitvallon/100-best-books/master/books.json):

{

   "author": "Dante Alighieri",

   "country": "Italy",

   "imageLink": "images/the-divine-comedy.jpg",

   "language": "Italian",

   "link": "https://en.wikipedia.org/wiki/Divine\_Comedy\n",

   "pages": 928,

   "title": "The Divine Comedy",

   "year": 1315

}

Objetivo:

Usted ha sido contratado para implementar herramientas de análisis de los datos de una base de datos de libros, que una vez deserializadas aparecen como listados de diccionarios de Python

[

   ...,

   {

   "author": "Dante Alighieri",

   "country": "Italy",

   "language": "Italian",

   "pages": 928,

   "title": "The Divine Comedy",

   "year": 1315

   },

   ...

]

La idea es que implemente tres diferentes funciones de una librería llamada “libros ”, tendiente producir estadísticas por diferentes criterios. Las funciones son:

- estadisticasAutor: A partir de una base de datos representada por un listado de diccionarios de Python, debe retornar un diccionario en donde cada llave es el nombre del autor (cadena de caracteres), y el valor es la cantidad de libros que escribió (valor entero).

- estadisticasFecha: A partir de una base de datos representada por un listado de diccionarios de Python, debe retornar un diccionario en donde cada llave es un año de publicación de los libros (valor entero), y el valor es la cantidad de libros que publicaron en dicho año (valor entero).

- estadisticasPais: A partir de una base de datos representada por un listado de diccionarios de Python, debe retornar un diccionario en donde cada llave es el país de origen de los libros, y el valor es un diccionario que tiene como llaves los idiomas de escritura de los libros y como valor la cantidad de paginas que componen esos libros.

Nota: El campo "language" en algunos registros viene compuesto (Ej. "French, English").  No obstante para el propósito del reto, debe ser tratado como un único registro, que producirá una sola llave dentro del diccionario de respuesta de idiomas por país.

Ejemplo:

import libros

db = [

  {

   "author": "Giacomo Leopardi",

   "country": "Italy",

   "language": "Italian, French",

   "pages": 184,

   "title": "Poems",

   "year": 1818

  },

  {

   "author": "Elsa Morante",

   "country": "Italy",

   "language": "Italian, French",

   "pages": 600,

   "title": "History",

   "year":    1974

  },

]

resultado = libros.estadisticasPais(db)

print(resultado)

El anterior programa imprimirá:  
{'Italy': {'Italian, French': 784}}

este programa no requiere entrada. Ni generará salida. Se requiere que el tripulante genere un archivo con el nombre libros.py y que se respeten los nombres de las funciones dadas y sus parámetros.

Este esqueleto del archivo libros.py puede ser utilizado como referencia.

Nota:  No dejar de incluir el encabezado de codificación ( # -\*- coding: utf-8 -\*-)

# -\*- coding: utf-8 -\*-

def estadisticasAutor(db):

   """

   Retorna un diccionario con los nombres de los autores de los libros como llave y

   para cada uno la cantidad de libros correspondientes (valor entero)

   """

   resultado = {}

   #implementar lógica

   return resultado

def estadisticasFecha(db):

   """

   Retorna un diccionario que tienen como llaves los años en que se publicaron libros (valor entero) y  
   para cada año la cantidad de libros correspondientes (valor entero)

   """

   resultado = {}

   #implementar lógica

   return resultado

def estadisticasPais(db):

   """

   Retorna un diccionario con los paises como llave y como valor un diccionario en donde las llaves son los  
   idiomas de escritura y el valor es la cantidad total de páginas de dichos libros .

   """

   resultado = {}

   #implementar lógica

   return resultado