

EJERCICIO VIDEOJUEGOS (Basado en el Primer Parcial de FP-IS - 2018/19)

Autores: Antonia M Reina, José C Riquelme (última modificación 16/11/2019)

Disponemos de un conjunto de datos con información sobre ventas de videojuegos. Los datos se encuentran almacenados en un fichero en formato CSV codificado en UTF-8. Cada registro del fichero ocupa una línea y contiene los datos correspondientes a un videojuego: posición en el ranking de ventas, nombre del videojuego, plataforma, año de salida al mercado, género, compañía distribuidora, volumen de ventas en Norteamérica, Europa, Japón y otras zonas, y volumen global de ventas (el volumen de ventas viene dado en millones de copias). Estas son las primeras líneas del fichero (<https://bit.ly/2NQLwm8>) :

Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
1	Wii Sports	Wii	2006	Sports	Nintendo	41.49	29.02	3.77	8.46	82.74
2	Super Mario Bros.	NES	1985	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24
3	Mario Kart Wii	Wii	2008	Racing	Nintendo	15.85	12.88	3.79	3.31	35.82
4	Wii Sports Resort	Wii	2009	Sports	Nintendo	15.75	11.01	3.28	2.96	33

De los siguientes ejercicios los señalados con puntuación fueron los que se pusieron en el examen.

1. lee_juegos(nom_fichero): lee el fichero de entrada y lo almacena en una lista de tuplas. (1 Punto)
2. existe_mas_ventasJP(lista_juegos): cuántos videojuegos han tenido más ventas en Japón que en Norteamérica.
3. juegos_distribuidora_anyo(lista_juegos, publisher, anyo): obtiene una lista con los nombres de los juegos de una distribuidora dada en un año dado. (1 Punto)
4. num_distribuidoras(lista_juegos): obtiene el número de compañías distribuidoras distintas.
5. juego_mas_antiguo(lista_juegos): cuál es el videojuego más antiguo del fichero. Si hay empate en el año de publicación devuélvelos todos.
6. genero_mas_presente(lista_juegos): obtiene el género de juego que más veces se repite en el fichero.
7. genero_mas_ventas(lista_juegos): obtiene el género con el global de ventas mayor.
8. num_juegos_palabra(lista_juegos, palabra): devuelve el número de juegos que contienen una determinada palabra en el título.
9. mayor_dif_NA_EU(lista_juegos): cuál es el nombre del videojuego que mayor diferencia de ventas tiene en Europa frente a Norteamérica (a favor de Europa).
10. ventas_por_año(lista_juegos): devuelve una lista de tuplas formadas por un año y el global de ventas de ese año para los videojuegos que salieron ese año. Ordenada de mayor a menor volumen de venta.
11. dicc_porcentaje_ventasJP_por_año (lista_juegos): devuelve un diccionario con las claves formadas por los años de salida de un videojuego y los valores el porcentaje de las ventas en Japón frente al global.

12. `incremento_ventas(lista_juego)`: devuelve una lista ordenada cronológicamente con los incrementos en porcentaje de ventas globales de los videojuegos que se publicaron un año con respecto al anterior.
13. `juego_mas_ventas_globales_saga(lista_juegos, saga)`: obtiene una tupla formada por las ventas globales, el nombre del juego y el año, del juego de la saga dada como parámetro que más ha vendido. Se considera que un juego pertenece a una saga si en su nombre aparece la palabra dada como parámetro. Por ejemplo, serán juegos de la saga *Pokemon* aquellos que tengan *Pokemon* en su nombre. (1,5 Puntos)
14. `dicc_ventas_por_zona(lista_juegos, anyo_inicial=None, anyo_final=None)`: crea un diccionario con el acumulado de ventas por zona. Las claves del diccionario serán: América, Europa, Japón y Otros, y los valores el total de ventas para esa zona de los años incluidos en el rango (`anyo_inicial`, `anyo_final`). Si `anyo_inicial` es `None`, se devuelven las ventas acumuladas hasta `anyo_final`. Si `anyo_final` es `None`, se devuelven las ventas acumuladas desde `anyo_inicial`. Si ambos son `None`, se acumulan las ventas de todos los años registrados. (1,5 Puntos)
15. `dicc_top_n_juegos_por_genero(lista_juegos, n=1)`: crea un diccionario cuyas claves son los géneros y cuyos valores son una lista con los `n` mejores juegos de cada género en formato tupla (posición, nombre). Se considera que un juego es mejor que otro si su posición en el ranking es menor. Si no se proporciona ningún número, entonces la lista solo tendrá el juego mejor. (1,5 Puntos)
16. `dibuja_ventas_por_zona(lista_juegos, anyo_inicial=None, anyo_final=None)`: dibuja un gráfico que muestra el total de ventas por zona (América, Europa, Japón y Otros) en un determinado rango (`anyo_inicial`, `anyo_final`). Si `anyo_inicial` es `None`, se devuelven las ventas acumuladas hasta `anyo_final`. Si `anyo_final` es `None`, se devuelven las ventas acumuladas desde `anyo_inicial`. Si ambos son `None`, se acumulan las ventas de todos los años registrados. (1,5 Puntos). Se usarán las siguientes sentencias:

```
fig = plt.figure(titulo)
ax = fig.add_subplot(111)
n_x = range(len(lista_zonas))
ax.bar(n_x, lista_ventas, width=0.8, align='center')
ax.set_xticks(n_x)
ax.set_xticklabels(lista_zonas, rotation='vertical')
plt.show()
```

donde `titulo` es una cadena con el título que aparecerá en la barra superior de la ventana en la que se muestra el gráfico, `lista_zonas` es una lista ordenada con el nombre de las zonas, y `ventas` una lista con las ventas asociadas a cada zona.

17. `distribuidora_mas_juegos_genero(lista_juegos, genero)`: obtiene el nombre de la distribuidora con más juegos del género dado como parámetro. (2 Puntos)
18. `juegos_distinto_ranking_EU_NA(lista_juegos, n)`: obtiene el conjunto de los videojuegos que están en el ranking de los `n` más vendidos en Norteamérica pero no entre los `n` más vendidos de Europa o al revés que sí están entre los `n` más vendidos de Europa y no en Norteamérica.
19. `juegos_mismo_ranking_EU_NA_JP (lista_juegos, n)`: obtiene el conjunto de los videojuegos que están simultáneamente entre los `n` primeros puestos de ventas en Europa, Norteamérica y Japón.
20. `primer_juego_distinto(lista_juegos)`: devuelve una tupla formada por los dos videojuegos de mayor venta que ocupan una posición distinta por ventas entre los rankings de Europa y Norteamérica.