

Proyecto Final

Fundamentos de Programación

Reportes: Juego de Tronos

Antecedentes

Se desea desarrollar un sistema para obtener estadísticas y reportes de la serie “Juego de Tronos” en Python. Para implementar este sistema se ha seleccionado un grupo de estudiantes con alto nivel de análisis y resolución de problemas, quienes deben aplicar uso de colecciones, estructuras de control, modularización y archivos para el desarrollo del sistema. Entrega: a las 11:55AM del día anterior a la sustentación vía SIDWEB.

Nota: Los grupos son asignados por el profesor.

Descripción

El proyecto final consiste en un sistema para generar reportes acerca de la serie Juego de tronos. Se debe modularizar el código de forma que las funciones estén guardadas en el archivo **funciones_got.py** y el programa principal debe estar en el archivo **principal_got.py**.

La información está almacenada en archivos de formato CSV, donde todas las líneas tienen el mismo formato específico. Debe leer los archivos y cargarlos en las estructuras aprendidas en clase y que faciliten la generación de los reportes y gráficos necesarios. Es obligatorio el uso de los módulos “pandas” y “matplotlib”.

Reutilice y/o modifique las funciones anteriormente programadas para el desarrollo de este proyecto.

1. Programa principal

En un archivo llamado **principal_got.py** debe escribir el programa principal. Este programa debe mostrar un menú con las siguientes opciones:

Menú principal - Reportes de Juego de Tronos

1. Importar datos de personajes, episodios y casas nobles.
2. Imprimir estadísticas para cada casa noble.
3. Imprimir episodio más visto en cada temporada.
4. Mostrar gráfico de barras de tiempo total que aparece cada personaje en la serie.
5. Mostrar gráfico de pastel de cantidad de espectadores totales en cada temporada.
6. Mostrar gráfico de barras de cantidad de personajes vivos vs muertos para cada casa noble.
7. Salir.

2. importarArchivos(personajes, episodios, casas)

Esta función recibe la ruta relativa de los archivos CSV para los personajes, episodios y casas nobles. Debe generar tres colecciones, una para cada archivo. Utilice pandas para las 2 primeras.

3. imprimirEstadisticasXCasa (personajes, casa= "Stark")

Esta función recibe la colección de los personajes y el nombre de una casa, por defecto es la casa Stark. Considere que hay personajes que son aliados a varias casas al mismo tiempo y debe ser considerado para todas ellas. Se debe mostrar las siguientes estadísticas para la casa recibida como argumento. Esta función debe además generar un archivo con las estadísticas, con el nombre "reporte<Casa>.txt", donde <Casa> debe ser cambiado por el nombre de la casa del argumento.

Imprimir en pantalla y guardar como reporte:

```
Estadísticas de la casa: "----"
-----
# personajes vivos
# personajes muertos
Tiempo total promedio: # que aparecen todos los miembros de esta casa en
todas las temporadas
Personaje más importante: # Nombre del personaje con mayor tiempo total en
todas las temporadas
```

4. imprimirEpisodiosMasVisto(episodios)

Esta función recibe una colección con todos los episodios. Se debe mostrar en pantalla el episodio más visto (tiene más espectadores) en cada temporada. Para el nombre del episodio solo debe mostrar los primeros 15 caracteres. Esta función además debe generar un archivo con el mismo formato mostrado en pantalla llamado "reporteEpisodios.txt".

Imprimir en pantalla y guardar como reporte:

Nro	Nombre	Rating	Espectadores (M)	Temporada
1	Winter Is Com.	8.5	2.22	1
...				

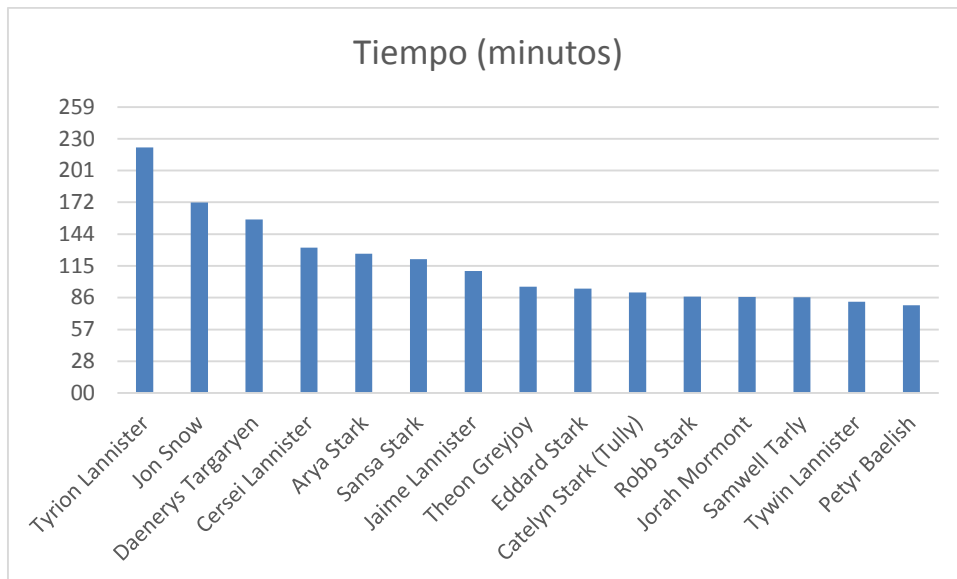
Nota: Debe utilizar la función format para dar el formato adecuado a la tabla.

<https://pyformat.info/>

5. mostrarTiempoTotal (personajes)

Esta función recibe una colección de personajes. Debe filtrar cuales son los 15 personajes más importantes (tienen más minutos en toda la serie) y graficar cuanto es el tiempo (minutos) que aparece cada personaje.

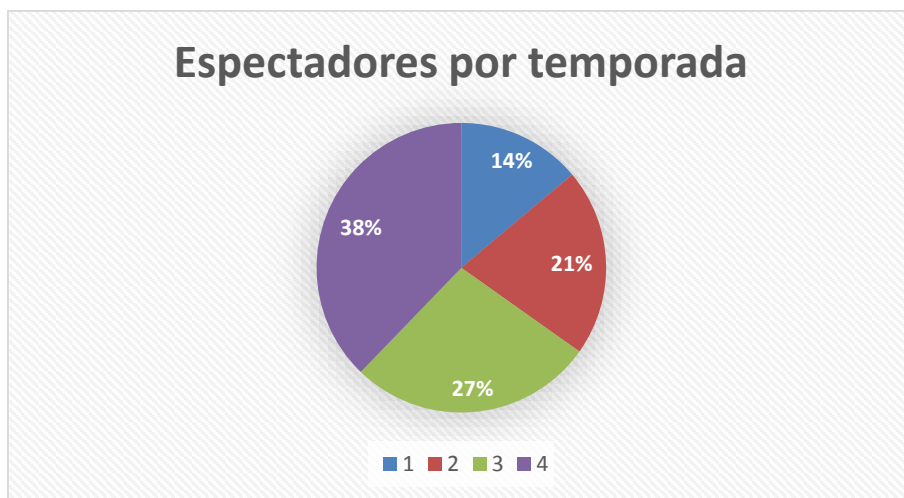
Gráfico esperado:



6. mostrarEspectadores(episodios)

Esta función recibe una colección con los episodios. Debe generar un gráfico de pastel con la cantidad total de espectadores en porcentaje que ha tenido cada temporada.

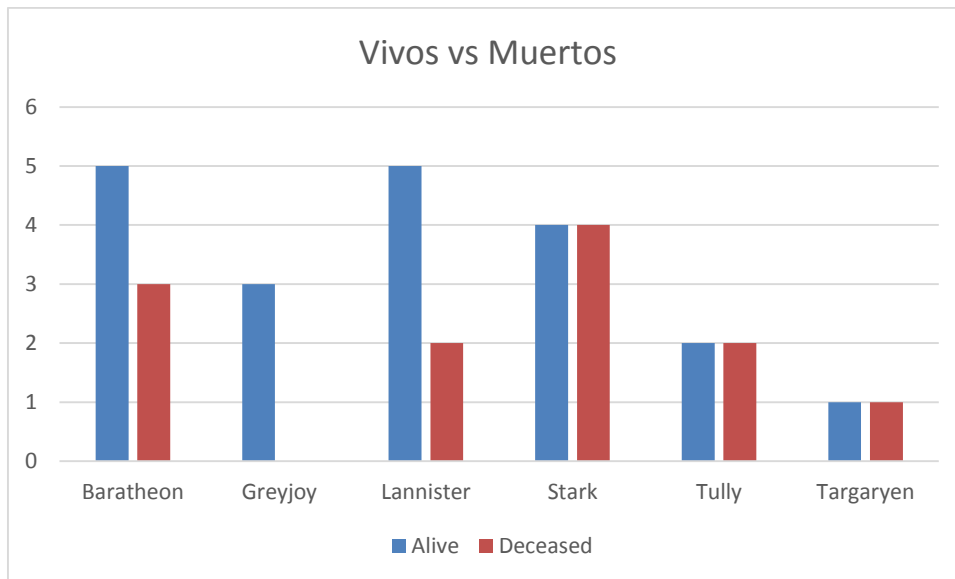
Gráfico esperado:



7. mostrarVivosvsMuertos(personajes)

Esta función recibe una colección con los personajes y debe generar un gráfico de barras de la cantidad total de personajes vivos y muertos por cada casa noble.

Gráfico esperado:



Subir a SIDWEB los archivos **funciones_got.py** y **principal_got.py** dentro de un archivo **ZIP o RAR**.

Rúbrica

Se calificarán los siguientes aspectos en porcentajes especificados.

1. Primer avance 25%:
 - a. Todos los literales valen lo mismo, 3 puntos, excepto el programa principal que vale 4 puntos.
2. Segundo avance 25%:
 - a. Modificar y/o reutilizar las funciones desarrolladas para el primer avance. 10 puntos.
 - b. Todos los literales de este avance valen lo mismo, 2 puntos. Excepto el programa principal que vale 3 puntos.
3. Entrega final 50%:
 - a. Para la entrega del proyecto final se deberá entregar por lo menos el 80% de las funcionalidades requeridas, caso contrario obtendrá una **calificación de cero**.
 - b. Uso de pandas y matplotlib. 6 puntos.
 - c. Todos los literales valen lo mismo. 7 puntos cada uno.