

Instrucciones

- Lea con detenimiento cada una de las actividades a realizar durante la experiencia.
 - Cree un archivo con extensión **.py** con lo desarrollado. El nombre del archivo debe tener el siguiente formato: **TEL101_CG_Nombre_Apellido.py** (Ej. TEL101_CG_Patricio_Olivares.py)
 - Enviar el archivo a través de la página de aula del ramo, sección “Control Global” hasta las 17:10:59 del presente día 18/08/2020.
 - Comente adecuadamente el programa, describiendo lo que hace.
 - Sea riguroso con las instrucciones de desarrollo.
 - ¡Éxito!
1. **Cine en su casa.** **IMDb** es una de las más populares fuentes de información sobre películas, series, etc. **Amazon**, la empresa dueña de **IMDb**, ha decidido realizar un análisis de sus datos de películas para mejorar su catálogo presente en su producto **Amazon Prime**. En este contexto, se ha solicitado sus servicios como programador.

Para esto, se le entrega a ud. un archivo con extensión **.csv** llamado **imdbMovies.csv** (presente en la página de **aula**), el cual contiene información de una parte de las películas presentes en sus registros. Este archivo contiene la siguiente estructura:

Título;Géneros;Código Lenguaje Original

donde

- **Título** representa el título de la película
- **Géneros** representa él o los géneros a los que pertenece la película en el siguiente formato: Género1 | Género2 | Género3 |...
- **Código Lenguaje Original** representa el código del lenguaje original de la película. Ej: es (Español), en (Inglés), it (Italiano), etc.

Ojo: Notar que los distintos campos están separados por **punto y coma (;)**.

Ej. de archivo **imdbMovies.csv**:

```
...
Mars Attacks!;Comedy|Fantasy|Science Fiction;en
Before Sunrise;Drama|Romance;en
Memento;Mystery|Thriller;en
Blade Runner;Science Fiction|Drama|Thriller;en
Hero;Drama|Adventure|Action|History;zh
Before Sunset;Drama|Romance;en
```

```

Miami Vice;Action|Adventure|Crime|Thriller;en
Open Water;Drama|Thriller;en
Raiders of the Lost Ark;Adventure|Action;en
The Elementary Particles;Drama|Romance;de
Indiana Jones and the Temple of Doom;Adventure|Action;en
Dirty Dancing;Drama|Music|Romance;en
...

```

1. (35pts) Cree la función ***getMoviesInfo(fname)*** que recibe como entrada una variable string ***fname***, con el nombre del archivo. La función retorna un diccionario de listas con la información contenida en el archivo. Las claves de este diccionario vienen dadas por un string con el nombre de la película. El valor asociado a cada llave será una lista que contiene en cada posición:

- Una tupla. Cada posición de la tupla representa uno de los géneros a los cuales pertenece la película,
- El código del idioma original de la película.

Si se ingresa en la variable ***fname*** algo distinto a un string, debe **levantar una excepción**. Ej.

```

>>> movieInfo = getMoviesInfo('imdbMovies.csv')
>>> print(movieInfo)
{...
'Thumbsucker': [('Comedy', 'Drama'), 'en'],
'The_Lost_Boys': [('Comedy', 'Horror'), 'en'],
'Ghost_World': [('Comedy', 'Drama'), 'en'],
'Games_of_Love_and_Chance': [('Romance', 'Drama'), 'fr'],
'Pieces_of_April': [('Comedy', 'Drama'), 'en'],
'Flatliners': [('Drama', 'Horror', 'Science_Fiction', 'Thriller'), 'en'],
'Parenthood': [('Comedy', 'Drama', 'Family'), 'en'],
...}

>>> movieInfo = getMoviesInfo(1234)
...
ValueError: Nombre de archivo no valido

```

2. (15pts) Cree la función ***getMoviesLang(movieInfo, lang)*** que recibe como entrada un diccionario de listas ***movieInfo*** con la información de las películas según el formato indicado en la pregunta 1, y un string ***lang*** que contiene el código de un idioma. La función retorna una lista con todas las películas cuyo idioma original es ***lang*** según la información contenida en ***movieInfo***. En caso de ingresar un código de idioma inexistente, se debe levantar una **excepción**

```

>>> mvs = getMoviesLang(movieInfo, 'es')
>>> print(mvs)
['Amores_perros',
'Talk_to_Her',
'Land_Without_Bread',
'All_About_My_Mother',
'Bad_Education',

```

```

    'Volver',
    'Live_Flesh',
    'Maria_Full_of_Grace',
    ...
]
>>> mvs = getMoviesLang(movieInfo, 'xyz')
...
ValueError: Codigo de idioma no valido

```

3. (20pts) Cree la función ***getMoviesGenre(movieInfo, genre)*** que recibe como entrada un diccionario de listas **movieInfo** con la información de las películas según el formato indicado en la pregunta 1, y un string **genre** que corresponde a un género cinematográfico. La función retorna una lista con todas las películas cuyo género es **genre** según la información contenida en **movieInfo**. En caso de ingresar un género inexistente, se debe levantar una **excepción**

```

>>> mvs = getMoviesGenre(movieInfo, 'Drama')
>>> print(mvs)
[...
'Rocky_IV',
'Rocky_V',
'Sweet_Sixteen',
'Houseboat',
'Shortbus',
'Lights_in_the_Dusk',
'Gleaming_the_Cube',
'The_Fountain',
...
]
>>> mvs = getMoviesGenre(movieInfo, 'Cauoi')
...
ValueError: Genero no valido

```

4. (25pts) + (5 pts) Cree la función ***plotInfoMovies(movieInfo, option, data)***, que recibe como entrada un diccionario de listas **movieInfo** con la información de las películas según el formato indicado en la pregunta 1, un entero **option** (el cual solo toma valores 1 o 2) y una lista de strings **data**. La función debe generar un gráfico de tortas, donde cada porción representa lo siguiente:

- Si **option=1**, **data** contiene una lista de códigos de lenguaje. El gráfico debe mostrar la proporción de películas hechas en los idiomas presentes en la lista **data**.
- Si **option=2**, **data** contiene una lista de géneros cinematográficos. El gráfico debe mostrar la proporción de películas en cada género cinematográfico presente en la lista **data**.

La función no retorna nada y debe **capturar cualquier excepción que se produzca**.

Adicionalmente, generando solo lo que se muestra en el ejemplo, ud. opta a un máximo de 25pts. Para optar al puntaje completo de esta pregunta (30pts) debe generar un gráfico **estético y comprensible**. Mejore el aspecto del gráfico según estime.

Ej:

```
>>> plotInfoMovies(movieInfo,1,['en','es','it','de'])
```

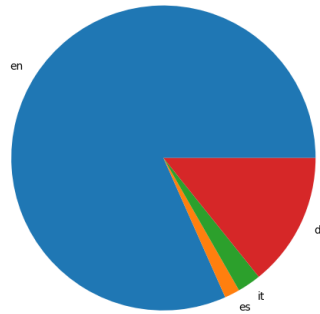


Figura 1: Gráfico generado por función `plotInfoMovies()`, con parámetro `option=1`

```
>>> plotInfoMovies(movieInfo,2,['Drama','Comedy','History'])
```

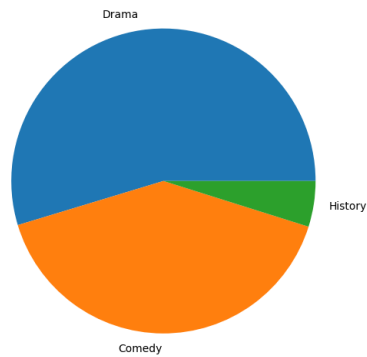


Figura 2: Gráfico generado por función `plotInfoMovies()`, con parámetro `option=2`

1 Referencias

Archivo .csv con datos de películas del sitio **IMDb** es una versión simplificada del archivo publicado en Kaggle en el siguiente enlace