

DESARROLLO DE SOLUCIONES WEB

RETO

En este reto desarrollarás un algoritmo que extraiga información de internet sobre distintas películas. Luego construirás algunas funciones que permitan hacer consultas sobre estas películas

PASOS A SEGUIR

1. Primero crea una cuenta en omdbapi

Web scraping normalmente se realiza mediante un servicio API. Este servicio es como un repositorio al que podemos hacerle consultas. En este caso, nos ayudaremos Open Movie Database (OMDB) quien nos da una API para realizar consultas. Para esto primero créate una cuenta en este servicio para esto entra a su página web y regístrate (<http://www.omdbapi.com/>).

Luego de registrarte se te entregará un CODIGO, guárdalo, este código es necesario para hacer las consultas más adelante.

2. Carga la información del archivo txt

Dentro de la carpeta de GitHub se encuentra un archivo llamado ***codigos_imdb.txt***, en este archivo se tienen 150 códigos de distintas películas. Estos códigos te ayudarán a buscar películas con el servicio API y poder obtener información de dichas películas.

Para esto, primero se te recomienda cargar el archivo, cada línea es un código distinto.

3. Usar el servicio API

Para usar el API de OMDB deberás de usar la librería `urllib` para hacer request de páginas web.

Para usar el API deberás de hacer un request a una dirección web a la cual deberás de cambiarle algunas cosas. La dirección web es:

`http://www.omdbapi.com/?apikey=[CODIGO_USUARIO]&`

Deberás de reemplazar el `CODIGO_USUARIO` por el código que te den al registrarte. Luego, también deberás de mandar en el request el código de la película.

Por ejemplo, la primera línea del archivo tiene el código: `tt0025316` y por ejemplo mi `CODIGO_USUARIO` es `8bcf12b7`. Entonces tendré que hacer el request a esta página web: `http://www.omdbapi.com/?apikey=8bcf12b7&i= tt0025316`

Al hacer este request obtendremos la siguiente información:

```
{ "Title": "The Shawshank Redemption", "Year": "1994", "Rated": "R", "Released": "14 Oct 1994", "Runtime": "142 min", "Genre": "Crime, Drama", "Director": "Frank Darabont", "Writer": "Stephen King (short story \ "Rita Hayworth and Shawshank Redemption\ "), Frank Darabont (screenplay)", "Actors": "Tim Robbins, Morgan Freeman, Bob Gunton, William Sadler", "Plot": "Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.", "Language": "English", "Country": "USA", "Awards": "Nominated for 7 Oscars. Another 19 wins & 29 nominations.", "Poster": "https://ia.media-imdb.com/images/M/MV5BMDFkYTc0MGEtZmNhMC00ZDIzLWFmNTEtODM1ZmRlYWwMwMWFmXkEyXkFqcGdeQXVyMTMxODk2OTU@._V1_SX300.jpg", "Ratings": [ { "Source": "Internet Movie Database", "Value": "9.3/10" }, { "Source": "Rotten Tomatoes", "Value": "91%" }, { "Source": "Metacritic", "Value": "80/100" } ], "Metascore": "80", "imdbRating": "9.3", "imdbVotes": "1,929,977", "imdbID": "tt0111161", "Type": "movie", "DVD": "27 Jan 1998", "BoxOffice": "N/A", "Production": "Columbia Pictures", "Website": "N/A", "Response": "True" }
```

4. El objetivo del reto

En primer lugar, tendrás que crear una “BD” es un archivo txt, donde cada fila sea cada uno de los requests. Es decir, deberás de tener información de 150 películas (dado que se tiene 150 códigos en el archivo txt)

Luego, sobre este archivo deberás de crear algunas funciones que permitan hacer algunas consultas:

- La primera función tendrá solo un argumento que es el valor. Este valor es la calificación que se la da a la película (en el json esta variable se llama “Value”). El resultado de la función será una lista con las películas que tengan más o igual de valor que el parámetro elegido
- La segunda función tendrá como argumento dos parámetros. El primer parámetro será una fecha inicial y el segundo una fecha final. La función retornará el número de películas que se encuentran dentro de estas fechas

CONSIDERACIONES

- Grupos de 2 personas (o de 1 persona)
- Mandar sus soluciones en código máximo el sábado a las 11:59 pm