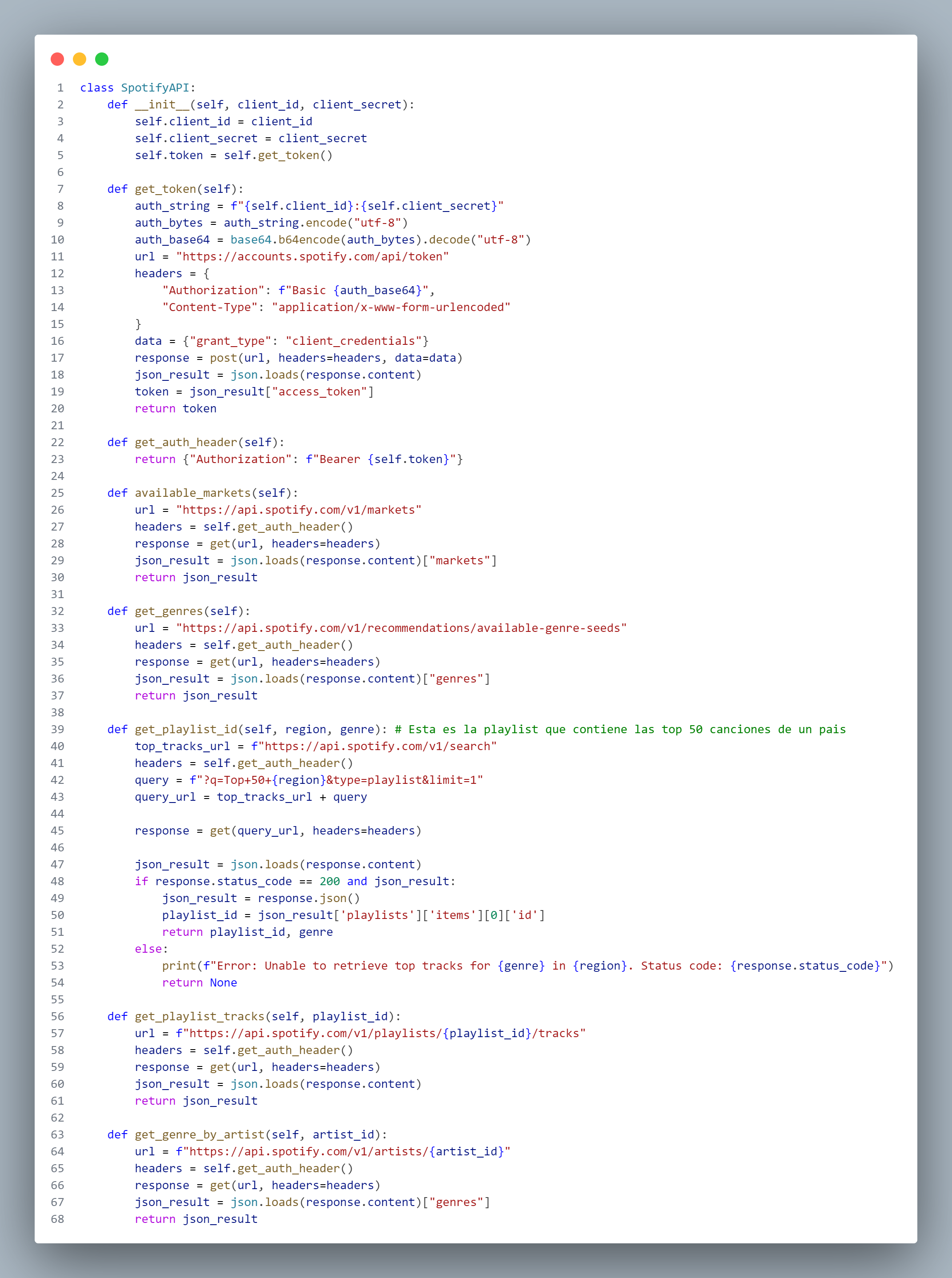
Aquí se define la clase encargada de realizar la adquisición de información y validación de las credenciales de la api. Se define tanto su constructor como los métodos necesarios y los endpoints relacionados a cada una de las consultas. Para obtener toda la información relacionada con los endpoints se realizo el uso de la documentación proporcionada por la api de Spotify en su sitio web. Se hicieron varios tipos de consultas, las relacionadas con el artista, las relacionadas con la playlist (top 50 canciones de cada pais), las canciones de dicha playlist por pais , una relacionada con los mercados disponibles, para que el usuario inicial sea capaz de seleccionar una región especifica, las canciones relacionadas con la playlist por país y por ultimo una consulta que relacionara el artista con su genero especifico.



Tras realizar el correspondiente diseño para cada método de dicha clase, se propone utilizar ciertos métodos, para obtener la información relevante de la siguiente manera.



Se le solicita al usuario la inserción de una de las regiones a analizar mediante el método input.

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Tras haber seleccionado y validado una región especifica se accede a la playlist de dicha región, en donde se relacionan las 50 canciones más importantes de esa región para ese día.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Si la lista existe entonces se realiza la obtención de la información mediante el uso de la clase creada previamente.

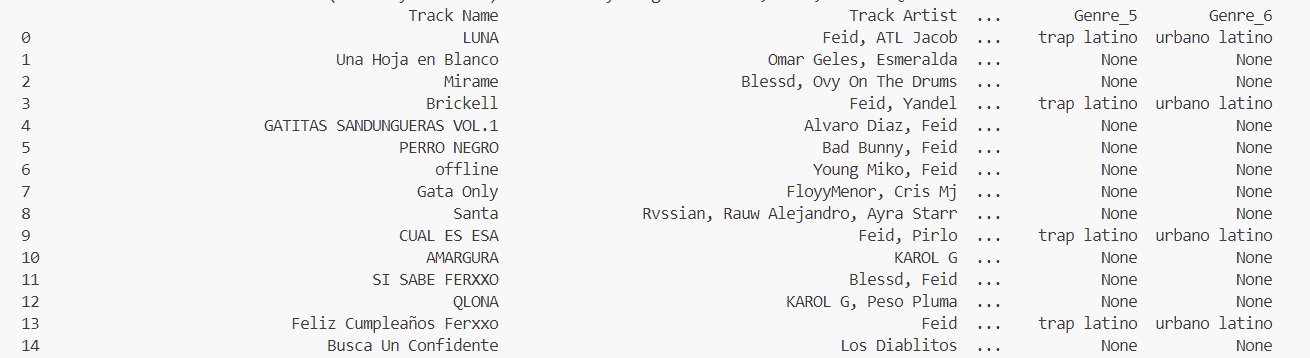
Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

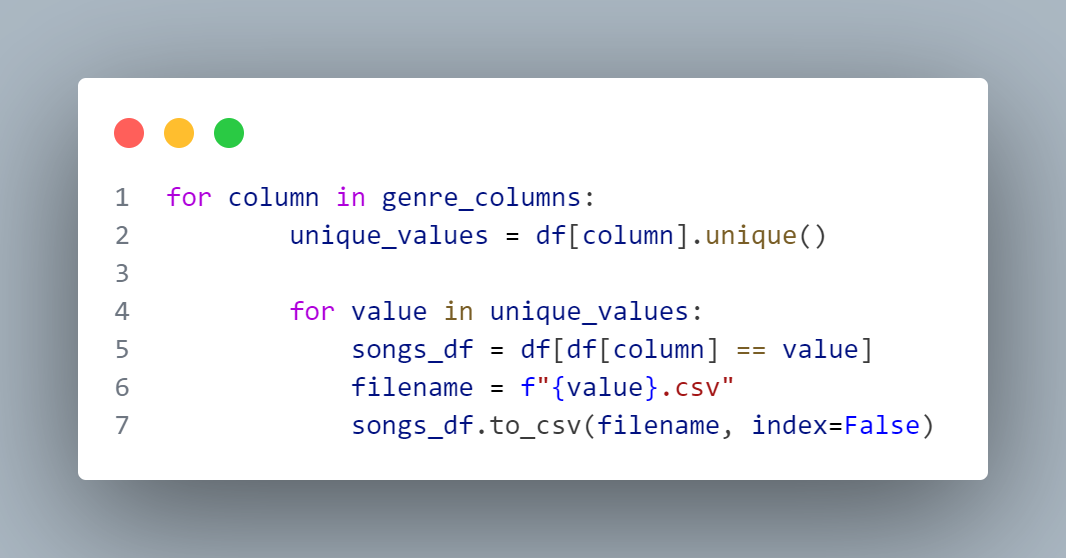
Se obtienen las canciones de dicha lista, el artista y el genero de dicha canción, para posteriormente ser almacenado en un dataframe, el cual contendrá la información previamente mencionada, LA CUAL SE VE DE LA SIGUIENTE MANERA.



Al observar la columna Track Genre, se analiza que esta contiene un array el cual me va a complicar un poco al momento de hacer la agrupación por género, así que la solución para este problema especifico fue separar por comas los valores de dicha columna y crear nuevas columnas que me permitieran almacenar el valor de los géneros de manera independiente. Esto con el fin de agrupar por las nuevas columnas y así obtener los géneros independientes a los que pertenece la canción.

Al realizar dicha separación se obtuvieron 6 columnas que para este caso de ejemplo son las necesarias para que funcione en la región de Colombia dicho script **(una solución un poco más robusta seria observar el tamaño máximo del array de track genre y con base en ese tamaño crear la cantidad de columnas.**) Obteniendo de esta manera un nuevo dataframe que me permite agrupar por las 6 columnas de genre, como se puede ver en la imagen siguiente.   
  


Lo ultimo que se hace es recorrer todas las columnas de géneros, obtener los valores únicos de dichas columnas, recorrer por cada valor único, obtener todas las filas que tengan ese valor único y crear varios archivo en donde se almacenen dichas canciones.

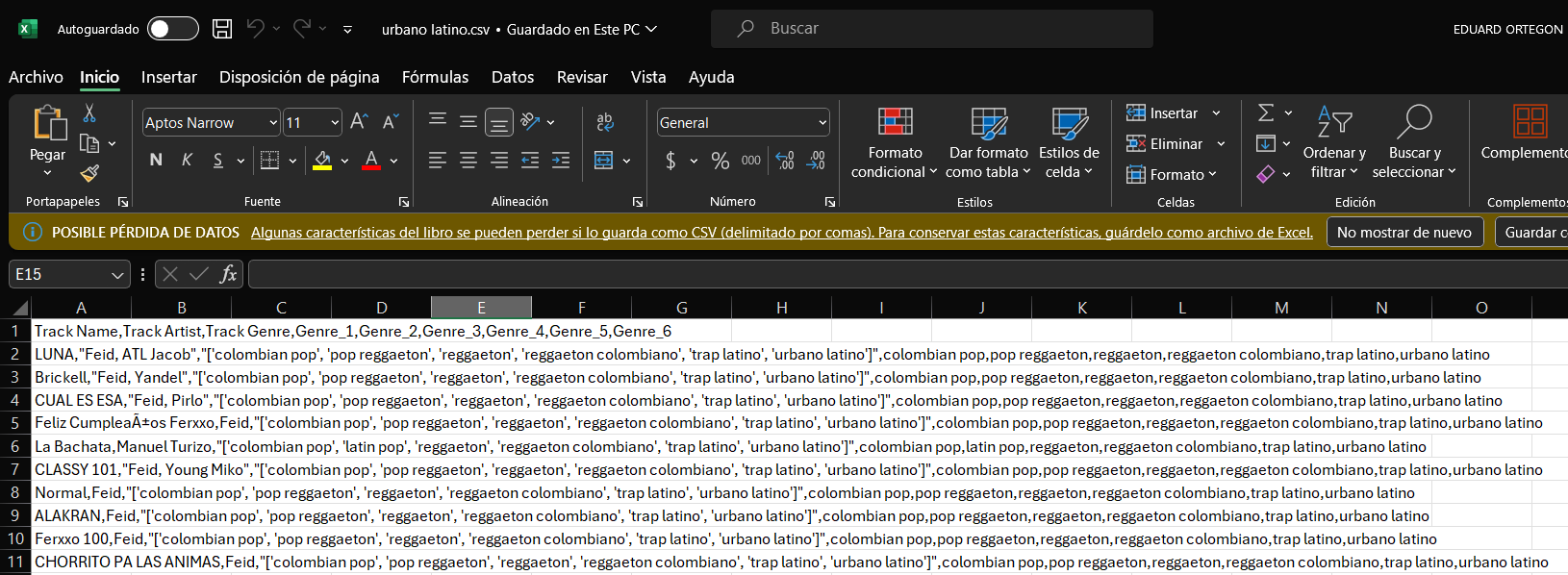


Para al final obtener un conjunto de archivos csv en donde se ven todas las canciones correspondientes a un genero especifico separadas por la región seleccionada por el usuario. **Si quisiéramos que el sistema agrupara por región de manera automatica sin la intervención del usuario final, se podría usar el método available markets de la clase Spotify API para obtener los diferentes mercados y así iterar por todos los mercados obteniendo las top 50 canciones de cada mercado y separándolos por archivos. Demostrando de esta manera la escalabilidad del código diseñado.**

**De esta manera se demuestra el funcionamiento del script diseñado.**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente



Captura de pantalla de computadora

Descripción generada automáticamente con confianza media

Interfaz de usuario gráfica

Descripción generada automáticamente