



Spotify a SQLite

USING SPOTIFY API TO POPULATE A SQLITE DATABASE

| DESARROLLO DE APLICACIONES PARA CIENCIA DE DATOS |
| 2º | GCID | ESCUELA DE INGENIERÍA INFORMÁTICA | UPGC |

Resumen

Este trabajo se centra en el uso de Java para manejar dos recursos muy comunes, las APIs y las Bases de Datos. Podemos entender la API como un código que indica a las aplicaciones cómo pueden mantener una comunicación entre sí. Estas reglas permiten que los distintos programas mantengan interacciones. Igualmente, una base de datos es un programa capaz de almacenar gran cantidad de datos, relacionados y estructurados, que pueden ser consultados rápidamente de acuerdo con las características selectivas que se deseen. En Java, para poder manejar estas bases de datos se hace uso de JDBC.

En cuanto al funcionamiento del programa, la clase Main contiene la lista de los artistas de los cuales vamos a pedir información. Mientras que las clases de SpotifyAccesor y SpotifyAuthorization se encargan de conectar con Spotify y autorizar tus pedidos a la web mediante tu Client ID y Client Secret. Asimismo, DataBaseManager se encarga de crear la base de datos y de insertar los datos en dicha base. Finalmente las clases de Album, Artist y Track tienen la función de solicitar a la API de Spotify los datos de dichas categorías y en almacenarlos en mapas, siendo todos estos datos de tipo String, pues de lo contrario complicaba el uso de mapas.

Índice

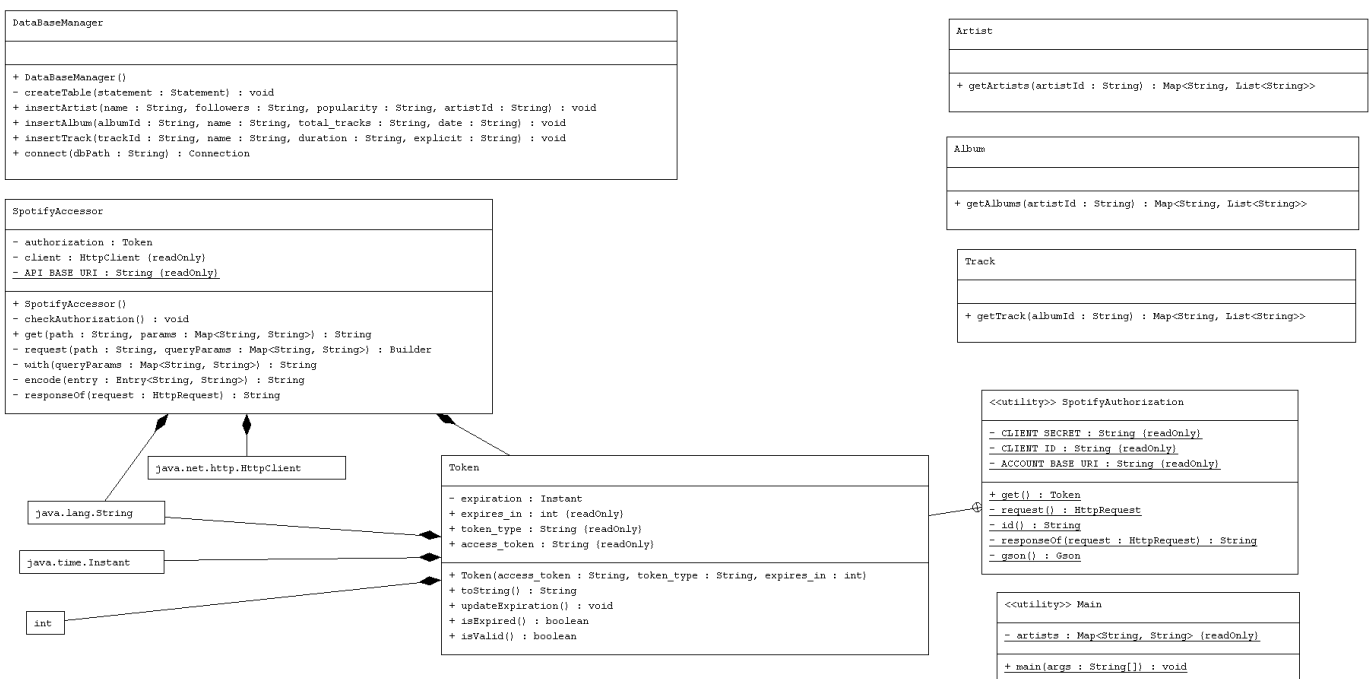
· Resumen.....	1
· Recursos Utilizados.....	3
· Diseño.....	3
· Conclusiones.....	4
· Líneas Futuras.....	4
· Bibliografía.....	5

Recursos Utilizados

Para la realización de este trabajo se ha usado principalmente el entorno de desarrollo de *IntelliJ IDEA Community Edition 2022.2.1* mientras que para su documentación se ha usado el software de Microsoft Word 2019. Finalmente para subir el programa se ha hecho uso de GitHub.

Diseño

A lo largo de este trabajo se ha tenido en mente el principio de “Responsabilidad Única”, donde una clase solo debe tener una única responsabilidad y solo debería tener una razón para cambiar.



Conclusiones

Por un lado, este trabajo me ha ayudado a entender mejor como funciona una API y todo el proceso que hay detrás de estas. De igual manera, he entendido como crear bases de datos y realizar todo tipo de operaciones en ellas a través de JDBC. Por otro lado, he visto que la manera en la que enfoqué este trabajo inicialmente no fue la mejor, haciendo uso de mapas pues me parecieron la mejor opción para empezar mientras entendía el trabajo. Más adelante comprendí que hubiera tenido mucha mayor facilidad a la hora de añadir los datos a la base de datos si hubiera usado las clases POJO, cosa que tendré en cuenta a partir de ahora.

Líneas Futuras

Hay varias maneras de las que se podría enfocar este proyecto a la hora de comercializarlo. En este caso me gustaría enfocararlo como una herramienta para trabajar con APIs, en este caso la Spotify API, sin que el usuario tenga que escribir el código. Esto podría realizarse con una interfaz que solamente pida al usuario los datos que quiere obtener de la API de Spotify y donde quiere almacenarlos. Todo esto funcionando en una página web en la que el usuario tendrá que dar su Client ID y Client Secret.

Bibliografía

- <https://developer.spotify.com/>
- <https://www.sqlitetutorial.net/>
- <https://www.javatpoint.com/sql-tutorial>
- <https://stackoverflow.com/>

Autor: Gabriel Felipe Bernal Pinto

Fecha: 09/11/2022

Version: 1.0