The *Spotify Web API* is a powerful tool that allows you to access and make changes to songs, albums, playlists, and other data objects stored in Spotify's database. The best part is, it's **completely free!**

In this tutorial, you will learn:

* How to install the API in Eclipse
* How to register your application

## Install the API

To use the Spotify Web API, We'll need a 3rd party library to help our application communicate with Spotify's servers. Luckily for us, Spotify has provided us with one [Here](https://github.com/thelinmichael/spotify-web-api-java). However, since the API library depends on other 3rd party libraries that we don't have, we must install it as a Maven dependency.

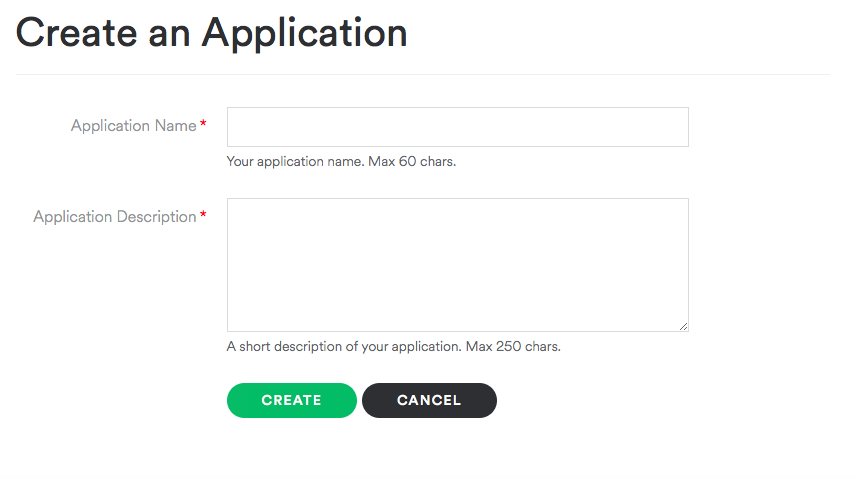
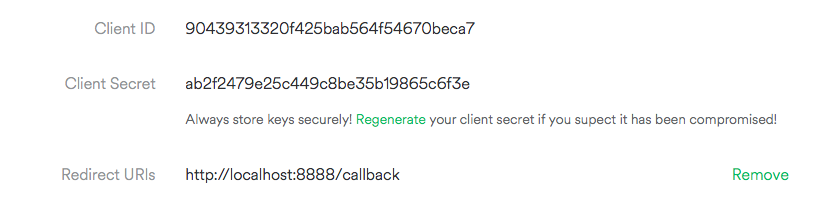
#### Installing the Maven Dependency

|  |  |
| --- | --- |
| Step | Screenshot |
| First, right click on your java project and select configure -> convert to maven project |  |
| Left click on the folder again and select maven -> add dependency A dialog box should prompt you for a **group id**, an **artifact id**, and a **version**. |  |
| Find the latest version of the library [here](https://oss.sonatype.org/#nexus-search;quick~spotify-web-api-java.). Select the file spotify-web-api-*version*.jar and copy the information into the dialog box in Eclipse |  |
| Click "Ok." The library binaries should appear in your project under **Maven Dependencies** |  |

## Register your application

In order to get faster request speeds or access any user-specific data, you will have to **register** your application with [Spotify's Developer Page](https://developer.spotify.com/my-applications/#!/applications). You'll need a spotify account (both free and premium accounts will work)

After creating an account, go to the *My Applications* page and select **"Create an App"**. After submitting a name and description for your application, you will be provided with a **Client ID** and a **Secret ID**. Store these keys somewhere safe, you will need them for the [Authorization Code Flow](https://developer.spotify.com/web-api/authorization-guide/#authorization-code-flow).

Set a **Redirect URI** if you plan on authenticating users. The link does not have to work, but the user sign-in page has to redirect to somewhere or it will cause an error.

## All Done!

You are now all set to start programming with the Spotify Web API! For documentation on how to use the API, check out the [readme](https://github.com/thelinmichael/spotify-web-api-java/#usage).