




Work on project. Stage 4/5: Processing links

Project: Music Advisor

Processing links

 Hard  2 hours  393 users solved this stage. Latest completion was **10 days ago**.

§1. Description

§2. GSON library

Since API basically responds in JSON format, we need to handle it somehow. Writing your own JSON parser will take a long time, so we will use the ready-made solution of the GSON library made by Google. The `build.gradle` file already has this dependency connected, see this line:

```
compile 'com.google.code.gson:gson:+'
```

For more information about JSON format, check out [the quackit tutorial](#).

GSON library provides different APIs to work with json. The simplest way to parse a json is using a `JsonParser` class: see [howtodo.injava.com](#) to learn more.

Here is a small example how to parse a simple json object:

```
{
  "name": "John",
  "address": {
    "city": "London",
    "plot": 12
  },
  "age": 23,
  "hobbies": ["music", "tech news", "blog"]
}
```

Code snippets

```
String json = "{\"name\":\"John\", \"address\":{\"city\":\"London\", \"plot\":12}, \"age\":23, \"hobbies\":"
            + "[\"music\", \"tech news\", \"blog\"]}";
JsonObject jo = JsonParser.parseString(json).getAsJsonObject();
String name = jo.get("name").getString();
int age = jo.get("age").getAsInt();

JsonObject addressObj = jo.getAsJsonObject("address");
String city = addressObj.get("city").getString();
int plot = addressObj.get("plot").getAsInt();

List<String> hobbies = new ArrayList<>();
for (JsonElement hobby : jo.getAsJsonArray("hobbies")) {
    hobbies.add(hobby.getString());
}
```

§3. REST principles

Since the Spotify API is based on simple REST principles, we must learn what REST means. Representational State Transfer (REST) is a software architectural style that defines a set of constraints to be used for creating web services.

It works over HTTP. The main concept of REST is resource: every URL defines its own resource and every HTTP method defines a verb you want to do with this resource.

Examples

/albums with the GET method should return representational state of all albums;

/albums with the POST method should create new resource over /albums URL;
/albums/5 with the PUT method should update information about the fifth album;
/albums/5 with the DELETE method should delete album number 5.

§4. Adding authorization header

Every request to Spotify API must contain a header "Authorization" with the value "Bearer + access_token". Extra information may be passed in query parameters (if it is a GET method) or in the request body (if it is a POST method).

Here is an example how to pass the `access_token` to the `HttpRequest` class:

```
HttpRequest httpRequest = HttpRequest.newBuilder()
    .header("Authorization", "Bearer " + accessToken)
    .uri(URI.create(apiPath))
    .GET()
    .build();
```

§5. What should you do

In this stage, you should replace your fake responses with the real ones. You should make requests to Spotify API and parse responses with the json parser.

Let's remember what information should be provided for each user request:

- `featured` — list of Spotify featured playlists with their links fetched from API;
- `new` — list of new albums with artists and links on Spotify;
- `categories` — list of all available categories on Spotify (just their names);
- `playlists C_NAME`, where C_NAME — name of category. List contains playlists of this category and their links on Spotify;

Full documentation how to work with api could be found at <https://developer.spotify.com/documentation/web-api/reference/>.
Object model: <https://developer.spotify.com/documentation/web-api/reference/object-model/>

To get all categories, use `https://api.spotify.com/v1/browse/categories`

To get a playlist, use `https://api.spotify.com/v1/browse/categories/{category_id}/playlists`

To get new releases, use `https://api.spotify.com/v1/browse/new-releases`

To get featured playlists, use `https://api.spotify.com/v1/browse/featured-playlists`

Pay attention to `playlists` action. Make sure that inside the request you send **category id, not category name!** Category names can contain spaces or other invalid URL symbols. So you should request category ids and names before the `playlists` request and find the id by category name. If the id format is correct but you cannot find it in the list of ids, print "Unknown category name".

In case of invalid category id (contains invalid URL symbols) in playlist request or other API error, the program should output the error message from the Spotify response. For example, if you get the response `{"error": {"status": 404, "message": "Specified id doesn't exist"}}`, you should print the following line: "Specified id doesn't exist".

Also, in this stage, you should read 2 server points from command line arguments:

- `-access` argument should provide authorization server path. The default value should be `https://accounts.spotify.com`
- `-resource` argument should provide API server path. The default value should be `https://api.spotify.com`

Make sure you replace constants to these argument values everywhere.

§6. Example

Below is an output example of the described program. Try to output all cases like in the example.

The greater-than symbol followed by a space (`>`) represents the user input. Note that it's not part of the input.

```
> new
Please, provide access for application.
> auth
use this link to request the access code:
https://accounts.spotify.com/authorize?client_id=a19ee7dbfda443b2a8150c9101bfd645&redirect_uri=http://localhost:8080&response_type=code
waiting for code...
code received
Making http request for access_token...
Success!
> new
NOT ALL HEROES WEAR CAPES
[Metro Boomin, Travis Scott, 21 Savage]
https://open.spotify.com/album/1zNr37qd3iZJ899byrTkcj

I Used To Know Her - Part 2 - EP
[H.E.R.]
https://open.spotify.com/album/46imFLgb9fR1Io6EoPYeQh

The Last Rocket
[Takeoff]
https://open.spotify.com/album/5XRCCUfwTLNQf1Dd9cfz4U

Interstate Gospel
[Pistol Annies]
https://open.spotify.com/album/0IXxmmlfSQxgJNwnNjHhgJ

El Mal Querier
[ROSALÍA]
https://open.spotify.com/album/355bjCHzRJztCzaG5Za4gq

I Found You (with Calvin Harris)
[benny blanco, Calvin Harris]
https://open.spotify.com/album/3jfsJj7tHJxk1a4i2KZt9F

Back to Life (from "Bumblebee")
[Hailee Steinfeld]
https://open.spotify.com/album/3sSEnG2RZGnZlTFwJwxPy4

> featured
Creativity Boost
https://open.spotify.com/playlist/37i9dQZF1DX56qfiUZBncF

Music for a Workday
https://open.spotify.com/playlist/37i9dQZF1DXcst4WKI8W8r

Dance Rewind
https://open.spotify.com/playlist/37i9dQZF1DX0BcQWzuB7ZO

Acoustic Throwbacks
https://open.spotify.com/playlist/37i9dQZF1DX8TvdYVZSYFY

Hit Rewind
https://open.spotify.com/playlist/37i9dQZF1DX0s5kDXi1oC5

Soul Coffee
https://open.spotify.com/playlist/37i9dQZF1DXaXdsfv6nvZ5

Workday Lounge
https://open.spotify.com/playlist/37i9dQZF1DWT5lkChsPmpy

> categories
Top Lists
Pop
Mood
Hip-Hop
Decades
Chill
Electronic/Dance
```

```
R&B
Party
Indie
Workout
Season's Greetings
Rock
Afro
Sleep
> playlists Sleep
Sleep
https://open.spotify.com/playlist/37i9dQZF1DWZd79rJ6a7lp

Songs For Sleeping
https://open.spotify.com/playlist/37i9dQZF1DWStLt4f1zJ6I

Peaceful Piano
https://open.spotify.com/playlist/37i9dQZF1DX4sWSpwq3LiO

Ambient Chill
https://open.spotify.com/playlist/37i9dQZF1DX30go9pFvBkY

Night Rain
https://open.spotify.com/playlist/37i9dQZF1DXbcPC6Vvqudd

> playlists unknown
Unknown category name.
> exit
```

[Report a typo](#)[⚡ See hint](#)[↩ Write a program](#)[Code Editor](#)[IDE](#)

CONNECTION STATUS

IDE / Checking the plugin's status

[Solve in IDE](#)[Look up solution \(💎 100\)](#)[Comments \(139\)](#)[Hints \(8\)](#)[Useful links \(5\)](#)[Solutions \(64\)](#)[Show discussion](#)