Problem description:

You will create a software that retrieves information about Star Wars characters. Your software will interact with a public API called swapi.co (“Star Wars API”).

In the swapi.co API you can access data about films and characters of the Star Wars movies. Each character has a species, for example Luke Skywalker is a *Human* and C-3PO is a *Droid*.

The software you create should expose an API that receives two parameters, a film ID and a character ID.

Given those two parameters you must consult the swapi.co API and return a list of the characters of the film you received as parameter that have the same species as the character you received as parameter.

Example:

Given you receive the parameters **film_id=1** and **character_id=2**

Sample swapi.co API calls:

Request	Response

GET: https://swapi.co/api/films/1	{
	"title": "A New Hope",
	"episode_id": 4,
	"director": "George Lucas",
	"characters": [
	"https://swapi.co/api/people/1/",
	"https://swapi.co/api/people/2/",
	"https://swapi.co/api/people/3/",
	"https://swapi.co/api/people/4/",
	"https://swapi.co/api/people/5/",
	"https://swapi.co/api/people/6/",
	"https://swapi.co/api/people/7/",
	"https://swapi.co/api/people/8/",
	"https://swapi.co/api/people/9/",
	"https://swapi.co/api/people/10/",
	"https://swapi.co/api/people/12/",
	"https://swapi.co/api/people/13/",
	"https://swapi.co/api/people/14/",
	"https://swapi.co/api/people/15/",
	"https://swapi.co/api/people/16/",
	"https://swapi.co/api/people/18/",
	"https://swapi.co/api/people/19/",
	"https://swapi.co/api/people/81/"
	]
	}

GET: https://swapi.co/api/people/2	{
	"name": "C-3PO",
	"height": "167",
	"mass": "75",
	"hair_color": "n/a",
	"skin_color": "gold",
	"eye_color": "yellow",
	"birth_year": "112BBY",
	"gender": "n/a",
	"homeworld": "https://swapi.co/api/planets/1/",
	"films": [
	"https://swapi.co/api/films/2/",
	"https://swapi.co/api/films/5/",
	"https://swapi.co/api/films/4/",
	"https://swapi.co/api/films/6/",
	"https://swapi.co/api/films/3/",
	"https://swapi.co/api/films/1/"
	],
	"species": [
	"https://swapi.co/api/species/2/"
	]
	}

GET: https://swapi.co/api/species/2	{
	"name": "Droid",
	"classification": "artificial",
	"designation": "sentient"
	}

Note that C-3PO (/people/2) is of the Droid (/species/2) species. In this case you must return all Droids on the movie “A New Hope” (/films/1).

Expected Response:

["C-3PO", "R2-D2"]

Restrictions:

- You must use Java 8 or superior
- Your application must use Maven or Gradle
- The API your application exposes must use REST principles
- Your API must have the following endpoint: /api/jdtest
- Your API must receive parameters as query parameters.
  - Ex: ?film_id={ID}&character_id={ID}
- Do not use any already available library to connect to swapi.co, you are required to implement the API calls yourself.

Nice to have but not mandatory:

- Use a web framework like Spring or similar
- Implement security on your API
- Implement validation and error handling on your API
- Handle concurrent API calls
- Use a version control system
- Implement an UI