

Assignment 7

Due Date: May 2, 2023 at 7:00 PM

Overview

This assignment will be worth **9%** of your total grade. The goal of this project is to integrate multiple topics in the course to create a web scraper. If you use the ES6 import statement **instead** of the CommonJS require for node modules taught during lecture, you will lose 10%.

Specifications

For this assignment, you will need to understand the requests/Axios, cheerio, and nodemailer modules to parse an HTML webpage and email specific parts to an end-user.

Artist Email-Scraper

Create a file called artists.js (or a node project) that will scrape the provided url: <http://www.popvortex.com/music/charts/top-rap-songs.php> and send an email of all artists & songs that are sung by the specified artists given via the command line.

Requirements:

- You must read in the 'from', 'to', 'sender_email', and 'sender_password' from the credentials.json file. This does not need to be submitted in Bb, but when grading, I should be able to put in my own credentials.json file and should run smoothly. You will lose points if you do not use the correct JSON keys or if it does not read the values from credentials.json
- The specified artists must be read in via the command line
- If no artist(s) are specified **do not** send an email
- If the specified artist(s) are not found on the website, **do not** send an email
- If the specified artist(s) are found on the website:
 - Send an email from an email account of your choice, to an email account of your choice (when I am testing this, I will input my own accounts for this portion)

- The subject will be: “Your artist(s) are: ” followed by a list of the artist(s). Be sure to format this portion correctly i.e. if the user specifies *node artists Drake Migos xxxtentacion* your subject should be: **Your artists are: Migos, Drake, and xxxtentacion**
- The email format should be the artist in **bold** and the song name in *italics*. Features may be in the song name, so that is fine to leave artists that are featured there, see below

Things to note:

- Be careful when making too many requests to a website, there is a possibility of being rate-limited
- Only worry about the top 25 songs, the website is dynamic and sometimes has more or less songs depending on the day.
- Don't worry about case, it's up to the user to ensure spelling and proper casing
- Don't worry about artists that have more than one word in their name i.e. Cardi B and Post Malone. Although we love them, it makes it tricky when reading from the command line
- If the artist is included as a feature, please include that as a song i.e. if searching for Migos, please include the track - **Gucci Mane: *I Get The Bag* ft. Migos**
- Feel free to create a test Gmail account as a sender, it makes life much easier just so you don't have to fiddle with your security settings
- Remember to enable 2-factor for the email, generate an app password (name it whatever you'd like), and use that app password as the sender_password in your file

Submission

Please submit the following on Blackboard:

- artists.js

```
let jsonData = require('./credentials.json');
console.log(jsonData.from);
console.log(jsonData.to);
console.log(jsonData.sender_email);
console.log(jsonData.sender_password);
```

Figure 1: example credentials.json being read in

Your artists are: DaBaby and Drake

Jal Irani <nodemailertowson@gmail.com>

to me ▼

Drake: *What's Next*

Yung Bleu: *You're Mines Still (feat. Drake)*

Megan Thee Stallion: *Cry Baby (feat. DaBaby)*

DaBaby: *Masterpiece*

DaBaby: *ROCKSTAR (feat. Roddy Ricch)*

Pop Smoke: *For The Night (feat. Lil Baby & DaBaby)*

Drake: *Wants and Needs (feat. Lil Baby)*

SpotemGottem: *Beat Box 3 (feat. DaBaby)*

Figure 2: For Input: node artists.js DaBaby Drake

```
Jals-MacBook-Pro:Desktop jalirani$ node artists.js
You did not specify an artist(s)
Jals-MacBook-Pro:Desktop jalirani$
```

Figure 3: If no artist is specified