Afleveringsopgave

Cleaning Hogwarts Data

You have decided to get a head start on the large Hogwarts Studentlist assignment, so you are going to build a *vertical prototype* to test the requirements for the data supplied by the administration:

https://petlatkea.dk/2021/hogwarts/students.json

and you are shocked and appauled. Apparently the data-entry assistant - *you are told his name is Grawp* - isn't the most systematic or consistent, so there are lots of extra spaces in the student's names, missing or wrong capitalization, even of the house-names, and very inconsistent use of full names with or without middlenames, sometimes without last names at all, sometimes with extra nick names. At least everything is spelled correctly though.

When you asked the administration if they could provide a better version, they laughed, pointed to the forrest, and suggested you went and asked Grawp yourself - and encouraged you to bring a cross-bow "just in case".

So you have to write a program to clean the data. First you just build a prototype to test how they should be cleaned - later you might rework that prototype into the final product.

The Student object

Rather than using objects directly from JSON, as you have usually done, you are going to create completely new objects, with cleaned data from the JSON, and store those objects in a global array.

You will have to design a new Student object, that will contain all the needed data, in an orderly fashion.

You decide that it should at least contain:

- First name
- Last name
- Middle name (if any)
- Nick name (if any)
- Image/photo filename
- House

You design the prototype for the object, and then build code to read each JSON-object, and create a new Student object, and populate it with cleaned data from the JSON-object.

Store all the created student objects in a global array, and simply use `console.table` to show all the objects and their properties.

Cleaning data

Splitting into parts

Since the JSON-data only contains the full name of each student, you need to write code that splits it into parts, capitalizes those parts correctly, and puts them into the newly created student object.

If a student doesn't have a middle name, the object should either have null or undefined for the middle name - you decide.

If a student has multiple middle names, you can combine them all into a single string, but remember to capitalize each one.

Nick-names are put in quotation marks in the JSON, but should just be plain text in the student object.

Make sure that there aren't any spaces around names.

Capitalization

Usually the first letter of each name should be upper case, and the remaining should be lower case.

However, names with a hyphen, must have the first letter after the hyphen capitalized as well.

The house names should also be proper capitalized.

Images

You can download the images from https://petlatkea.dk/2021/hogwarts/images.zip. You don't have to use the images in this version, but find a way to extract/calculate the image filename from the student name. You will find a pattern!

You will also find problems - some students don't have images, and some have a bit different pattern. Decide how your code should handle these special cases! Try to make it as generic (future-proof) as possible.

Displaying data

Since you are building a technical prototype, you don't care about looks or presentation - simply console.table the entire list of student-objects.

Hand-in details

One to throw away

Since this is a technical prototype, you might not want to put it in the same git repository as the rest of the Hogwarts assignment – or if you do, you should make a separate branch for it. However, there are no requirements to put this assignment in either git or github!

How

Upload the finished code to a folder on your own domain.

Hand in a link to the primary html file in that folder.

Feedback

No immediate feedback is planned, this exercise is as much a way of keeping active with the large assignment, rather than pushing everything to the end.
We can discuss ideas and problems in the following week, if you request it.

P.S.

Consider this assignment the first version of the cleaning proces - as you learn more, you'll find the need to improve/replace your code. So keep it around, but more as a reference, than to keep working on!