CASE STUDY

Gotta Process 'Em all!

You are working at professor Oak's Pokémon research lab in Pallet Town, and has been tasked with preparing some data about Pokémon for one of your investors (running a state of the art Pokémon research lab is not free!). This investor creates video games about Pokémon, and professor Oak's lab provides information about all Pokémon to the investor, in order for the games to be as close to reality as possible.

After a vicious ransomware attack, the investor has lost a lot of the data and rely on you to help them get the lost data back.

Luckily you have all the information stored in the PokéAPI!

The PokéAPI is a RESTful API of all known Pokémon.

To get information on a single Pokémon the following GET request can be used `https://pokeapi.co/api/v2/pokemon/<index>`. Replace `<index>` with an integer to get the Pokémon with that index number.

For instance, to get information about the Pokémon with index 42, the url would be `https://pokeapi.co/api/v2/pokemon/42`, and the response would be about Golbat, a poison flying Pokémon.

The reponse returned from querying the PokéAPI is formatted as JSON. The structure of the JSON can be seen in the PokéAPI Documentation. The documentation also contains information on all the many other endpoints of the PokéAPI.

The Task

The investor would like the following information from you:

Primary Requirements

1. The `name`, `id`, `base_experience`, `weight`, `height` and `order` of all Pokémon that appear in the any of the games `red`, `blue`, `leafgreen` or `white`.
2. The `name` of the `slot` 1 (and if available 2) `type` of each of the Pokémon's `types`.
3. The Body Mass Index of the Pokémon (**hint**: The formula for BMI is weight (kg) / height (m^2))
4. The first letter of names of the Pokémon should be capitalized.
5. The url of the `front_default sprite`.
6. Prepare the data in an appropriate data format. Consider if it should be multiple or a single file.

Bonus Requirements

1. Imagine the Pokémon are subject to the GDPR. How would you pseudonymize the data to no longer make it re-identifiable?
2. Imagine you had to build a system in the cloud that would continiously deliver updates to the investor with updates about Pokémon. Draw an architecture for exposing new changes to the existing Pokémon to the investor.
3. Make a interactable dashboard using the data where users can get a detailed page about the Pokémon, including showing the image from the url of the `sprite`.
4. If a spark compatible framework was not already chosen for the primary requirements, consider how the code would change to be executed on a spark engine.

For the Interview

There is no time limit for solving the case. Use as much time and solve as much as you want to. The most important aspect is to have something to dicuss at the interview.

For the interview, please find a way to present the above in a meaningful way. This should include running some of your code as a live demo. You decide which parts.

Before the interview, please share the code and data by some appropriate means (e.g. a repository).