

Pokemon Gallery

Pokemon are creatures from a popular video game franchise. There are hundreds of Pokemon each with its unique number and name.

I want to display Pokemon from number 1 to 151 on a responsive webpage.

- Each Pokemon should show an image, number and name.
- Pokemon should be sorted by number.
- A maximum of 20 Pokemon should be shown at once.

Pokemon names API

The names of each Pokemon should be retrieved asynchronously using this RESTful API.

<http://pokeapi.co/api/v2/pokemon?limit=151>

Results are sorted by Pokemon number and this is the order they should be displayed on the page.

The API can potentially return Pokemon beyond #151. We do not require any Pokemon data beyond #151.

Search box

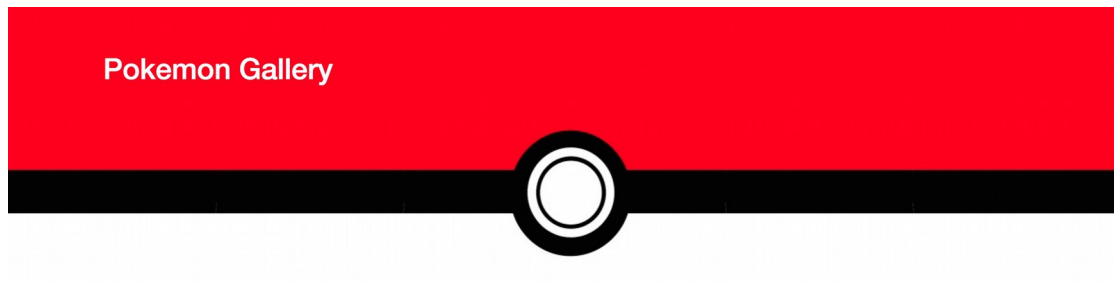
Provide a search box to filter Pokemon by name. The search term should be case insensitive and match any part of the name. For example, searching "Rat" will display Rattata, Raticate, Poliwrath, Dratini.

Pokemon images

The images of the 151 Pokemon are in the images directory. The filenames align with the Pokemon number, for example, #001 Bulbasaur uses 1.png.

Hero image and heading

Display the image "hero-background.jpg" at the top of the page. The image should take the full screen width and the heading should be shown in the foreground. (does not need to look exactly the same as the example).



Responsive grid

Breakpoints are defined as follows:

- **Mobile:** less than 768px
- **Tablet:** 768px and higher
- **Desktop:** 1200px and higher

On mobile screens display a list with 1 Pokemon per row.



#004

Charmander

On tablet screens and larger display a grid:

- **Tablet:** 2 Pokemon per row
- **Desktop and larger:** 4 Pokemon per row



#004

Charmander

Pagination

Provide a simple Previous / Next control at the bottom of the page so that a maximum of 20 Pokemon is displayed at once.

UI and user experience

The look and feel requirements have been purposely left vague. The UI just needs to be clean and easy to use. There is no need to make it look fancy.

Notes:

- You may use standard Javascript and CSS libraries. For example, JQuery, AngularJS, Bootstrap.
- Be prepared to talk about your solution should you progress to the next interview.
- There is no requirement to support legacy browsers.
- You should consider your code as production-ready so code quality and completeness is important.