JavaScript

Crear una Pokedex

Aquesta és la segona entrega pràctica del Mòdul JavaScript. Aquesta entrega té un pes del 50% de la PAC 3 (el 50% restant serà una prova teòrica de tipus test).

Introducció

L'objectiu d'aquesta pràctica es construïr una versió senzilla d'una Pokedex.

A la pàgina oficial de Pokemon es pot veure un exemple de Pokedex , tot i que l'objectiu és construïr una versió mínima d'això. Una Pokedex és un llistat de Pokemons des d'on es pot consultar informació detallada de cadascun d'ells.

Bàsicament, deixant de banda la temàtica de Pokemon, l'exercici tracta de consultar informació que ens arriba d'un recurs extern i un cop rebuda, maquetarla desde la nostra aplicació.

Tasques de l'exercici

Crear una web amb HTML, CSS i JS. A la pàgina principal (index.html) hi haurà una llista de Pokemons amb informació bàsica (nom i imatge), on al seleccionar-ne un, s'haurà de crear un sistema per mostrar més informació de Pokemon seleccionat (nom, imatge, valor d'atac, valor de defensa i

tipus). Si es vol mostrar aquesta informació en una nova pàgina, o sobreescriure la mateixa, s'haurà de crear un sistema de navegació per tornar a la pàgina inicial.

Com a alternativa, es pot mostrar la informació en una secció de la mateixa pàgina o en una finestra modal o pop up.

La pàgina ha de tenir un selector (radio button) que permeti escollir si es vol veure la web amb una paleta de colors clars o foscos. El funcionament s'especifica més avall.

Pàgina inicial (index.html)

- Consulta aleatòria a la API

Al carregar la pàgina inicial, s'ha de fer una consulta a la API de Pokemons (PokeAPI) i recuperar les dades de 10 Pokemons de forma aleatoria. Cadascun dels Pokemons s'ha de mostrar maquetat com una targeta o carta i ha de contindre la següent informació: nom i una imatge.

Aquesta informació es pot extreure de la API consultant l'ID del Pokemon. Per exemple, si es vol consultar el Pokemon amb ID = 1 s'ha de fer una consulta a la URL: https://pokeapi.co/api/v2/pokemon/1/

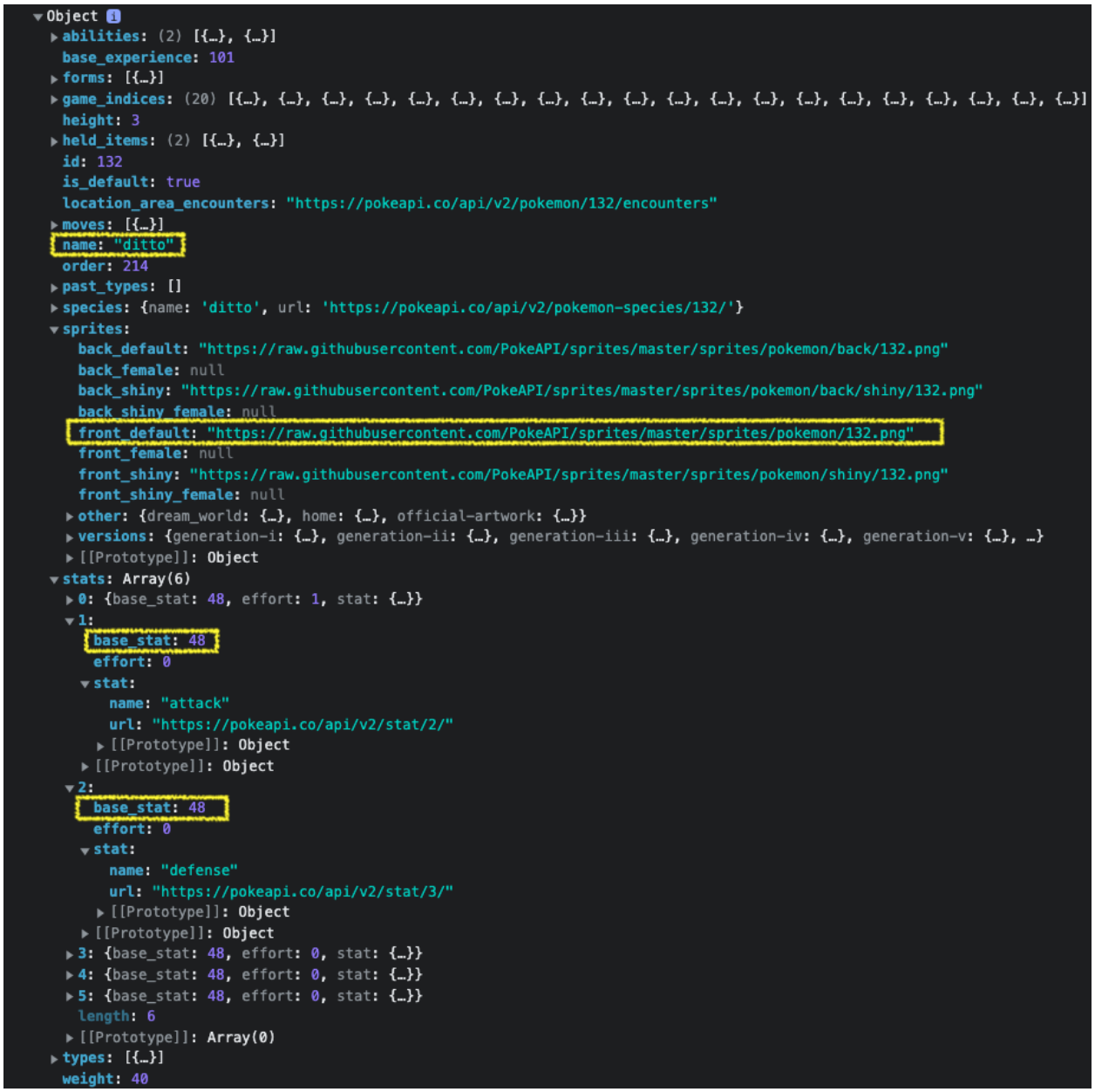
Per tant, s'haurà d'executar la crida 10 vegades, consultant cada cop la URL https://pokeapi.co/api/v2/pokemon/{ID} generant un número aleatori (random) a cada crida per obtenir un nou ID. Idealment s'hauria de controlar que no es generin elements duplicats.

La petició a la PokeAPI es pot fer utilitzant Fetch API

```
fetch('http://example.com/movies.json')
  .then(response => response.json())
  .then(data => console.dir(data));
```

Es pot obtenir la informació navegant per l'objecte que retorna la consulta.

Nom = `data.name`
Imatge = `data.sprites.front_default`
etc



- Informació ampliada de cada Pokemon

Les targetes de la pàgina principal també han de contenir un botó o un enllaç que permeti accedir a una pàgina amb informació més detallada del Pokemon. Aquesta informació es carregarà sobre la mateixa pàgina index.html, indicant a la mateixa URL un paràmetre amb l'ID del Pokemon que es vol consultar. Per exemple: index.html?pokeID=5.

Es pot fer com es cregui més convenient: esborrant/ocultant la informació de la pàgina i reprintant-la de nou, mostrar la informació en una finestra modal o popup o printant la informació en un contenidor (div o similar) que inicialment estigui buit i s'hi vagi printant la informació quan es vulgui.

Podeu trobar un exemple més avall a Continguts i recursos: exemple de paràmetres URL.

Al carregar la pàgina index.html s'ha de consultar si la URL conté el paràmetre pokeID. Si el conté, només s'ha de mostrar la informació del Pokemon en qüestió i un botó per tornar enrere. En cas contrari, ha de mostrar el llistat inicial de Pokemons.

Com a alternativa, també es pot mostrar la informació en una finestra moda o pop up, o en una secció específica de la mateixa pàgina.

La informació completa dels Pokemon que s'ha de mostrar en aquesta pàgina ha de ser:

Nom = `Objecte.name`
Imatge frontal = `Objecte.sprites.front_defau`
Imatge posterior = `Objecte.sprites.back_default`
Atac = `Objecte.stats[1].base_stat`
Defensa = `Objecte.stats[2].base_stat`
Tipus (poden ser varis) = `Objecte.types[x].type.name` : s'hauran de mostrar tots els tipus

- Filtre de cerca

La pàgina inicial també ha de tenir un filtre textual de Pokemons. Ha de ser un input de tipus "search" en el que, a mesura que es van escrivint lletres, només s'han de mostrar aquells Pokemon que el seu nom contingui el text escrit. Aquesta comprovació s'ha de fer per cada lletra que s'escrigui al filtre (event input).

Per exemple, si tenim les cartes: Charmander, Bulbasaur, Charizard i Blastoise, si s'escriu al filtre “cha”, només han de ser visibles les cartes Charmander i Charizard.

Barra de menú: selecció de tema clar/fosc

A la barra superior de menú hi ha d'haver un selector de tipus radio button amb 2 opcions: clar i fosc. Aquest selector, per defecte ha d'estar seleccionat amb l'opció de tema clar.

Els estils inicials de la pàgina i les targetes han de ser amb un color de fons clar amb una tipografia de color fosc. Si l'usuari selecciona l'opció de tema fosc, els estils de la pàgina han de canviar i els colors de fons i de les targetes han de ser foscos i la tipografia ha de ser clara (sempre a la inversa del color de fons per tal que hi ha un bon contrast i la legibilitat sigui accessible).

La selecció que fagi l'usuari s'ha de guardar al navegador com a Local Storage, de manera que quan l'usuari torni a carregar la web, el sistema recordi les seves preferències. Si havia seleccionat el mode fosc, la web s'ha de carregar directament en mode fosc. I el mateix pel mode clar. Si l'usuari no ha seleccionat cap preferència, s'haurà de carregar en mode clar.

Opcionalment es pot configurar perquè la web es carregui segons estigui configurat el navegador o el sistema operatiu de l'usuari. En aquest cas, s'haurà d'afegir una tercera opció al radio button que sigui "sistema" i que agafi els estils segons l'usuari tingui configurat el seu equip (clar o fosc).

A tenir en compte

No cal estar familiaritzat amb el món de Pokemon. S'ha decidit utilitzar aquesta API perquè és molt complerta i és gratuïta. És una API àmpliament utilitzada en exemples de programació de treball amb API, i és fàcil trobar exemples.

Objectius

El principal objectiu d'aquesta pràctica és treballar amb JavaScript i veure com es pot interactuar amb el DOM, creant i eliminant elements HTML i modificant i interactuant amb CSS.

Es veurà com treballar amb una API externa (https://pokeapi.co/), realitzant operacions de consulta, i s'aprendrà a interpretar les dades rebudes en format JSON.

Finalment es treballarà amb Local Storage, per aprendre a utilitzar el navegador per guardar informació sense haver de dependre d'una Base de dades.

Puntuació

Per avaluar aquesta pràctica es seguiran el següent criteri de puntuació:
1.5 punts: aplicació de bones pràctiques en l'ús de JavaScript, HTML i CSS
1.5 punts: ús de la API amb la funció fetch
1.5 punts: mostrar el llistat aleatori de Pokemons
1.5 punts: al seleccionar un Pokemon de la llista, mostrar la seva informació individual
1.5 punts: filtre de Pokemons; s'ha de poder filtrar els Pokemons de la llista pel seu nom
1.5 punts: funcionalitat dark/light mode
1 punt: estils CSS ben treballats.

Recursos

Generació de nombres aleatoris
https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Global_Objects/Math/random

Fetch API
https://developer.mozilla.org/es/docs/Web/API/Fetch_API/Using_Fetch

Consultar exemples de JavaScript al Github del curs
https://github.com/Aleix-Marti/curs-frontend/tree/main/JS

Exemple Pokedex
https://www.pokemon.com/us/pokedex

PokeAPI
https://pokeapi.co

Exemple CSS Flip Card
https://codepen.io/Aleix/pen/ozWBoG

Com crear un tema clar / fosc amb variables CSS
https://wordpress.tv/2020/01/22/aleix-marti-carmona-creando-un-dark-theme-con-variables-css-unete-al-lado-oscuro

Vídeo “dark-light-mode” de continguts i recursos de la PAC 3
https://github.com/Aleix-Marti/curs-frontend/tree/main/JS/dark-mode

7 Mètodes d'array en un minut
https://www.youtube.com/watch?v=96VjrPib6MA&ab_channel=midudev

Interacció DOM + JS
Vídeo “templates” de continguts i recursos de la PAC 3

https://github.com/Aleix-Marti/curs-frontend/tree/main/JS/templates

Fetch API
Vídeos “fetch-api-I” i “fetch-api-II” de continguts i recursos de la PAC 3
https://github.com/Aleix-Marti/curs-frontend/tree/main/JS/fetch

Modules
Vídeo “modules” de continguts i recursos de la PAC 3
Vídeos Modules-part01 i Modules-part02 de la carpeta del Drive de l’aula;
Codi dels exemples a:
https://github.com/Aleix-Marti/curs-frontend/tree/main/JS/modules

Repositori exemple Fetch API
https://github.com/LuichidevUOC/search-gifs

Exemples de càrrega de JavaScript
https://manzdev.github.io/script-type/

Exemple de paràmetres URL
https://github.com/Aleix-Marti/curs-frontend/tree/main/JS/url-params