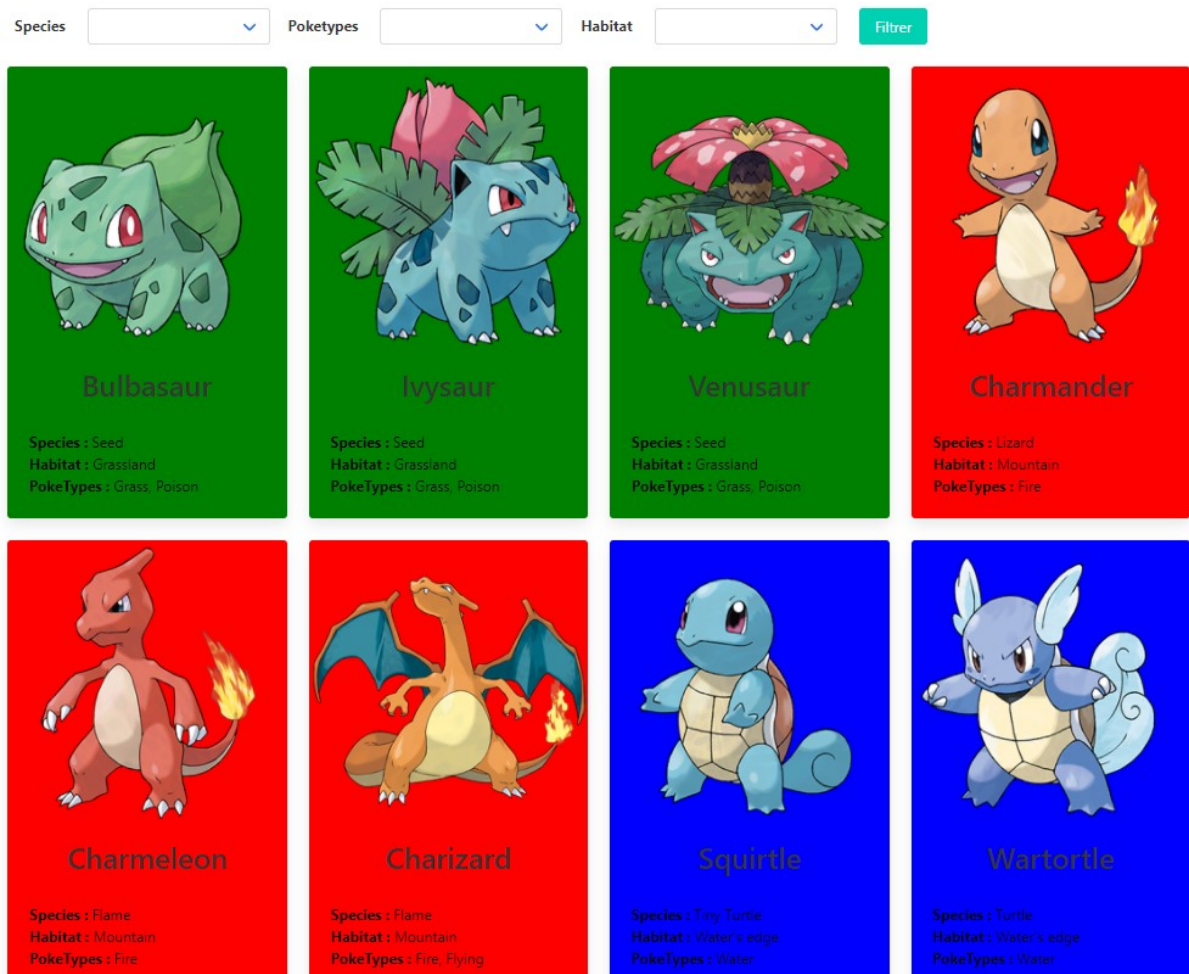


Objectifs

- Exploiter une API avec fetch.
- Exploiter une librairie CSS (bulma).
- Jeton d'accès et authentification.
- Appel à une méthode authentifiée.
- Voir le fichier TP1.gif
- À réaliser individuellement

Page Index

- Consiste à écrire un le code HTML et le code JavaScript nécessaire pour exploiter l'API <https://pokemonsapi.herokuapp.com/>.



- On affiche les Pokemons en 4 colonnes en mode desktop, 3 colonnes en mode tablette et en une seule colonne en mode mobile.
- Indiquer votre matricule (e1234567) au bas de la page ([footer](#)).
- Indiquer votre matricule (e1234567) dans le champ author du fichier package.json.
- Au chargement de la page, tous les Pokemons sont affichés.

- La liste de sélection des habitats affiche les habitats retournés par la méthode [/habitats](#) de l'api.
- La liste de sélection des poketypes affiche les poketypes retournés par la méthode [/poketypes](#) de l'api.
- La liste de sélection des espèces (*species*) affiche les espèces retournées par la méthode [/species](#) de l'api.
- Une zone de filtre permet de filtrer les Pokemons à l'aide de l'habitat, du poketype et/ou de l'espèce (*species*). Le bouton **filtrer** active le filtre.

Barre de navigation (pages index, connexion et favoris)

- Ajouter une barre de navigation à la page index.html (voir [navbar](#)) ainsi que les pages connexion.html et favoris.html.
- Au chargement de la page, lorsque l'utilisateur ou l'utilisatrice n'est pas connecté(e), la barre de navigation affiche:



- TP1 est un lien à la page index.html.
 - Connexion est un lien à la page connexion.html.
- Au chargement de la page, lorsque l'utilisateur ou l'utilisatrice est connecté(e) (voir plus bas), la barre de menu affiche:



- TP1 est un lien à la page index.html.
 - "Pokémons favoris" est un lien à la page favoris.html.
 - Déconnexion supprime le jeton (`sessionStorage.remove`) et déplace l'utilisateur à la page index.html (`window.location.href`).

Page Connexion

- Écrire la page connexion.html/connexion.js permettant à l'utilisateur ou l'utilisatrice d'entrer ses informations pour s'authentifier.

Email

e1234567@site.com

Password

.....

Connexion

Annuler

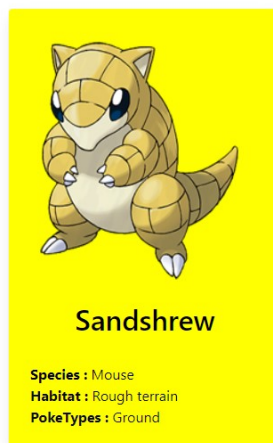
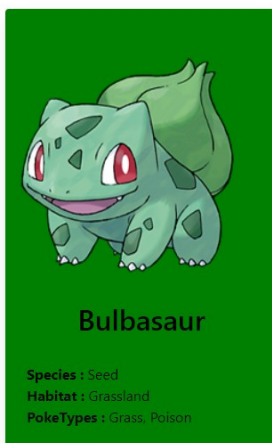
- Si la connexion est réussie, stocker le jeton d'authentification dans sessionStorage pour utilisation ultérieure (voir plus loin - page favoris.html) et ensuite rediriger l'utilisateur ou l'utilisatrice à la page de ses favoris.
- Si la connexion échoue, afficher un message d'erreur à l'utilisateur.
- Le bouton Annuler redirige l'utilisateur à la page index.html.
- Le bouton Connexion permet d'obtenir un jeton d'authentification en appelant la méthode [/auth/token](#) de l'API.
- Vous pouvez utiliser e1234567@site.com comme email et e1234567 comme password (où 1234567 est votre matricule étudiant) pour vous identifier à l'api.

Page Favoris

- Écrire la page favoris.html/favoris.js permettant à l'utilisateur ou l'utilisatrice de consulter ses Pokémon favoris.

TP1 Pokémon Favoris

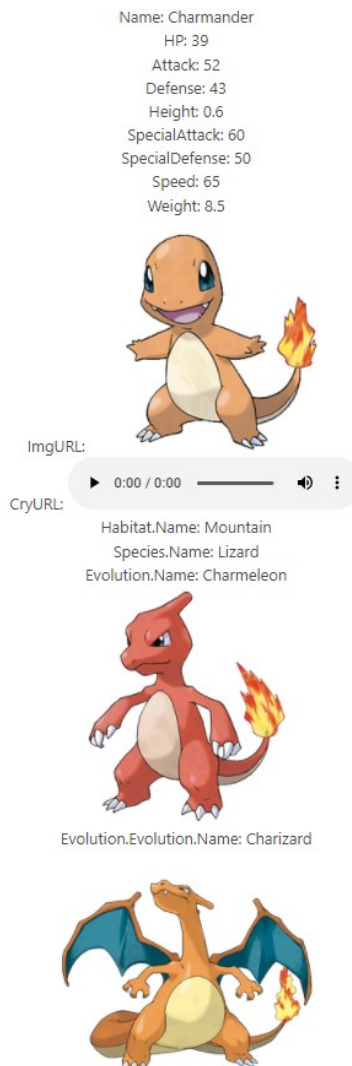
Déconnexion



- La méthode [/favorites](#) est utilisée pour obtenir les Pokémon favoris de l'utilisateur ou l'utilisatrice présentement connecté(e). C'est une méthode qui requiert l'utilisation d'un jeton d'authentification.
- La méthode [/favorites](#) retourne les données en utilisant le même format que la méthode [/pokemons](#).
- Au chargement de la page, si l'utilisateur ou utilisatrice n'est pas connecté(e), rediriger l'utilisateur ou l'utilisatrice vers la page connexion.html.

Détails

- Lorsqu'un Pokémon est cliqué dans la page index, la page details.html est appelée en spécifiant le pokemonId en paramètre (details.html?pokemonId=X).
- La page détails affiche toutes les informations concernant un Pokémon en exploitant la méthode </pokemon?pokemonId=X> où X est un pokemonId valide.



- Attention, ce ne sont pas tous les Pokémon qui possèdent une évolution.
- Pour les utilisateurs authentifiés, la page détails affiche aussi un bouton "ajouter aux favoris" ou un bouton "supprimer des favoris" selon le cas. Ces boutons permettent, respectivement, d'ajouter le Pokémon aux favoris de l'utilisateur ou de supprimer le Pokémon aux favoris de l'utilisateur.
- Pour déterminer si un Pokémon fait partie des favoris de l'utilisateur, vous devez utiliser la méthode de l'API </favorite?pokemonId=X> en "GET".
- Pour ajouter un Pokémon aux favoris de l'utilisateur, vous devez utiliser la méthode de l'API </favorite?pokemonId=X> en "POST".
- Pour supprimer un Pokémon aux favoris de l'utilisateur, vous devez utiliser la méthode de l'API </favorite?pokemonId=X> en "DELETE".

Commun.js

- Créer le fichier commun.js contenant :
 - La fonction `afficherPokemons`, qui génère le code permettant l'affichage d'une liste de Pokémons reçu en paramètre.
 - Ajoutez le code nécessaire pour exporter la constante et la fonction.
- Dans les fichiers `index.js` et `favoris.js`, importez et utilisez la fonction `afficherPokemons` pour réaliser l'affichage des Pokémons.
- Placez toute autre fonction utilisée dans plusieurs pages dans le module `commun.js`.