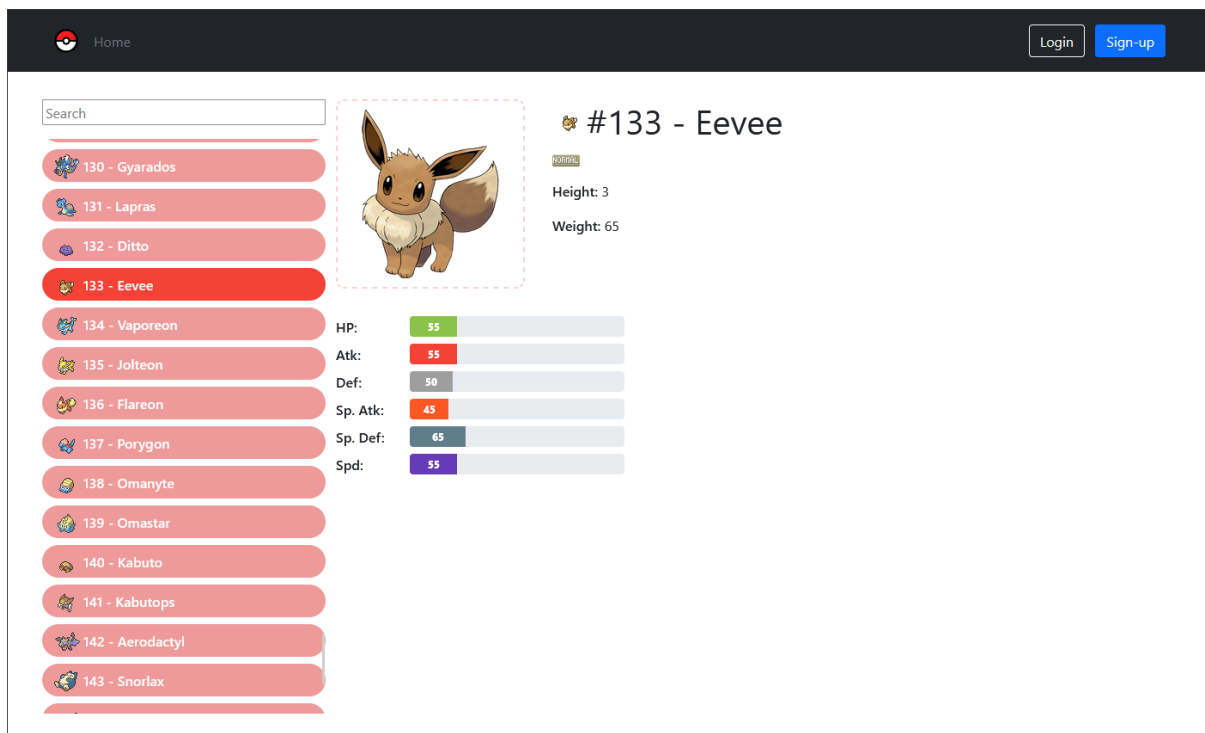


# TD EVALUE – POKEDEX

Le but de ce TD évalué est de réaliser une application Pokédex basée sur une API fournie gratuitement par <https://pokeapi.co/>. Pour ceux qui ne seraient éventuellement pas familier avec le concept, il s'agit essentiellement d'une encyclopédie de Pokémon. Pour faire simple, nous allons que gérer uniquement l'affichage des 151 premiers Pokémon. Ce sujet ne va pas rentrer dans trop de détails en ce qui concerne ces créatures et devrait être abordable même si vous n'y connaissez rien. Mais en cas de soucis, n'hésitez pas à poser vos questions.

Vous aurez à réaliser une page de vue générale présentant votre Pokédex, ainsi qu'un petit formulaire d'inscription. Voici à quoi ressemble l'application :



Vous partez comme d'habitude d'une base en HTML/CSS et vous devez construire à partir de celle-ci les fonctionnalités décrites dans ce document. Vous n'avez théoriquement pas besoin de toucher au HTML/CSS (mais ce n'est pas interdit).

## I/ MODALITES DE DEPOT ET PRECISIONS DE NOTATION

### A) RENDU DU TD ET MODALITES

Vous aurez 2 Google Forms à remplir dans le cadre de cette évaluation

1<sup>er</sup> formulaire : <https://forms.gle/EaedV6bzKqc6VYF7A>

Ce formulaire se compose d'un dépôt de .zip avec votre TD en l'état (peu importe si il fonctionne ou non) et de 2 questions théoriques. **Il doit être envoyé pour 12h00**

Le .zip reste important. Bien qu'il ne soit pas évalué directement (c'est bien votre rendu final qui sera comptabilisé), vous serez pénalisés si son avancement est insuffisant.

2<sup>ème</sup> formulaire : <https://forms.gle/r8QgB9AySBaqAUgg6>

Ce formulaire contient uniquement un dépôt de fichier .zip. Vous devez soumettre votre TD en état « finalisé ». **Il doit être envoyé pour le vendredi 1<sup>er</sup> avril à 23h59 au plus tard.**

## B) REALISATION DU TD ET PRECISIONS DE NOTATION

Chaque tâche spécifiée dans le TD est importante à sa manière. Mais vous remarquerez que certaines tâches, généralement plus complexes, sont préfixées par le texte « *POUR ALLER PLUS LOIN* » et sont écrites en italique. Le but de ces questions est de vous pousser à faire plus. Elles ne vaudront pas autant de points que les autres questions, mais seront tout de même nécessaires pour atteindre la note maximale.

## II/ LA SOURCE DE DONNEES

### A) L'API A UTILISER

Comme spécifié dans l'en-tête. Vous utiliserez l'API fournie par pokeapi. Vous pouvez accéder à la documentation de cette API ici : <https://pokeapi.co/docs/v2>. Elle est détaillée et permet d'accéder à beaucoup d'informations au besoin.

Regardez le détail si vous voulez. Mais essentiellement, vous aurez uniquement besoin d'utiliser les deux routes suivantes :

<https://pokeapi.co/api/v2/pokemon?limit=151&offset=0>

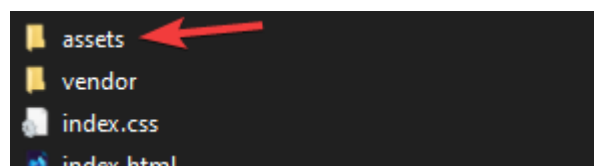
Cette première route va retourner les 151 premiers Pokémon présents dans la DB de l'API. Elle servira à peupler votre liste à gauche sur l'écran principal.

[https://pokeapi.co/api/v2/pokemon/\\${idPokemon}/](https://pokeapi.co/api/v2/pokemon/${idPokemon}/)

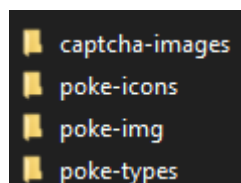
Cette deuxième route permet de récupérer les informations sur un Pokémon en particulier. Il faut juste remplacer dans l'URL son identifiant.

### B) LES FICHIERS FOURNIS

Vous aurez dans votre fichier .zip initial un certain nombre de fichiers à disposition. Pas de fichiers JS cette fois-ci, juste des fichiers images que vous devrez utiliser dans votre projet :

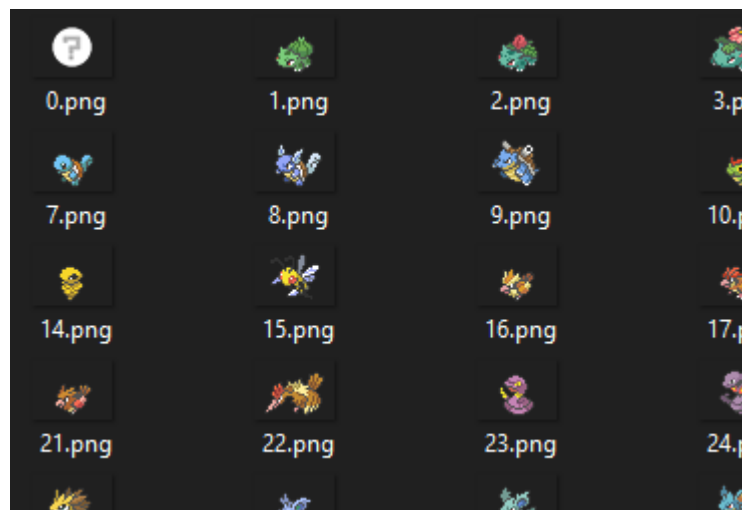


Tout se trouve dans le dossier « assets ». A l'intérieur, vous trouverez les dossiers suivants :

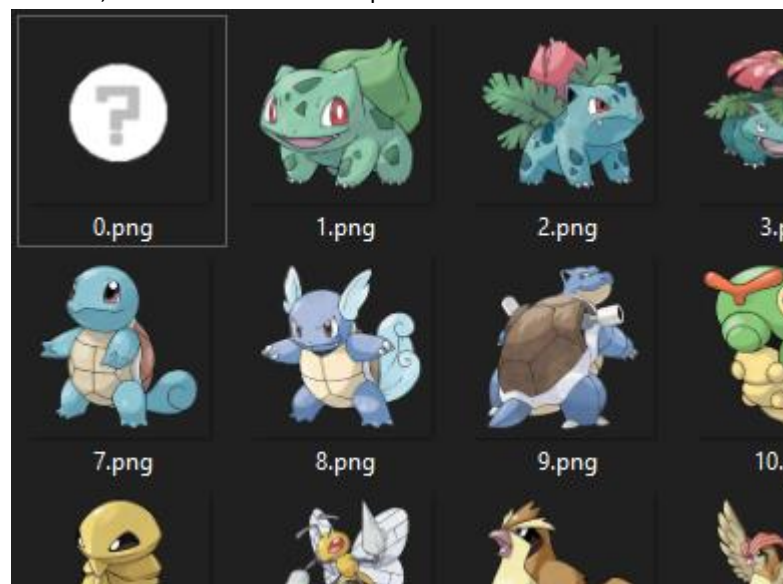


- Le dossier « captcha-images » est utilisé dans le fichier « signup.html » et contient les images à utiliser dans l'élaboration d'un captcha pour votre formulaire d'inscription

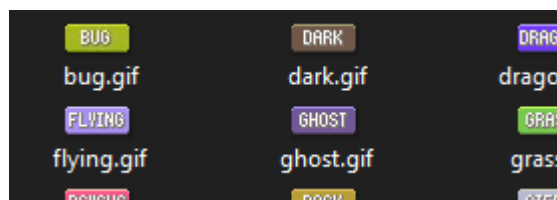
- Le dossier « poke-icons » contient les images des Pokémon en version « icône ». Le nom de fichier correspond à l'identifiant :



- Le dossier « poke-img » contient aussi des images de Pokémon mais en version « grand format ». Comme pour les icônes, le nom de fichier correspond à l'identifiant :



- Enfin, le dossier « poke-types » contient des images par rapport aux types de Pokémon. Leur nom de fichier match ce que vous avez en retour dans l'API :

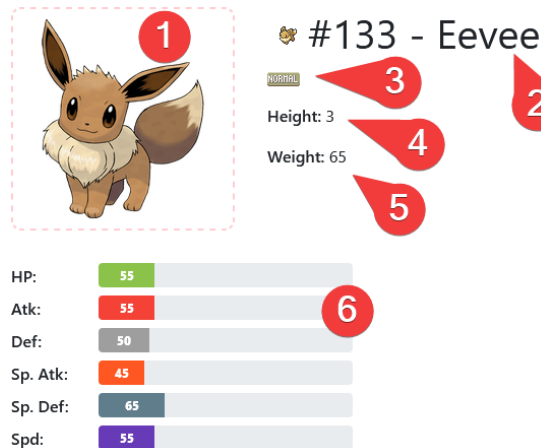


Avec tous ces fichiers, vous devriez avoir ce qui est nécessaire niveau images pour réaliser ce TD

### III/ TACHES A REALISER

#### A) LA PAGE "POKEDEX"

- Au chargement de la page, vous devez récupérer la liste des 151 Pokémons via l'API et les charger dans votre liste latérale
- Au clic sur un des éléments de la liste, vous devez charger les informations sur le Pokémon que vous avez cliqué et les afficher dans le volet de droite. Toutes les informations sont disponibles dans le retour de l'API. Prenez soin d'update les endroits indiqués ci-dessous :



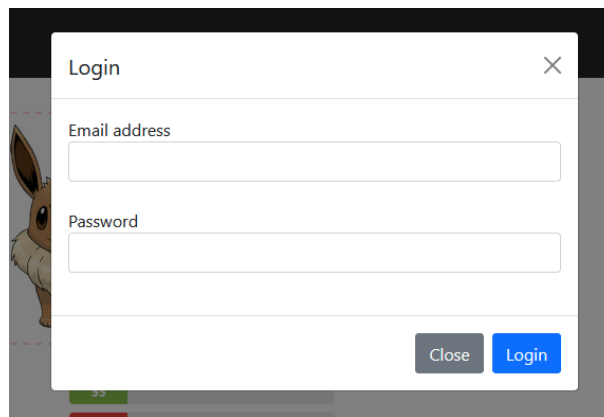
- Au chargement de la liste des Pokémons, sélectionnez par défaut le premier Pokémon de la liste
- **POUR ALLER PLUS LOIN :** Votre application pourra accepter un paramètre GET (dans l'URL) pour inclure une ID de Pokémon dedans. C'est cette ID qui sera chargée au lancement de l'application. Pensez aussi à changer l'URL à la sélection d'un Pokémon. Le but est de pouvoir partager un lien vers un Pokémon en particulier.
- **POUR ALLER PLUS LOIN :** Il y a un input texte au-dessus de la liste des Pokémons. A la saisie, vous devez uniquement afficher les Pokémons dans la liste qui correspondent à la recherche.

#### B) LA MODALE DE LOGIN

Dans la page HTML, il y a un bout de code correspondant à une modale de login, cachée par défaut via son attribut style :

```
</main>
<div
  class="custom-modal"
  id="login-modal"
  tabindex="-1"
  style="display: none"
>
  <div class="modal-dialog">
    <div class="modal-content">
      <form>
        <div class="modal-header">
          <h5 class="modal-title">Login</h5>
          <button type="button" class="btn-close"></button>
        </div>
        <div class="modal-body">
          <div class="mb-3">
            <label for="email" class="form-label">
              Email address
            <input type="text" class="form-control" id="email" />
          </div>
          <div class="mb-3">
            <label for="password" class="form-label">
              Password
            <input type="password" class="form-control" id="password" />
          </div>
        </div>
        <div class="modal-footer">
          <button type="button" class="btn btn-secondary">Close</button>
          <button type="button" class="btn btn-primary">Login</button>
        </div>
      </form>
    </div>
  </div>
</div>
</div>
<script src="index.js"></script>
```

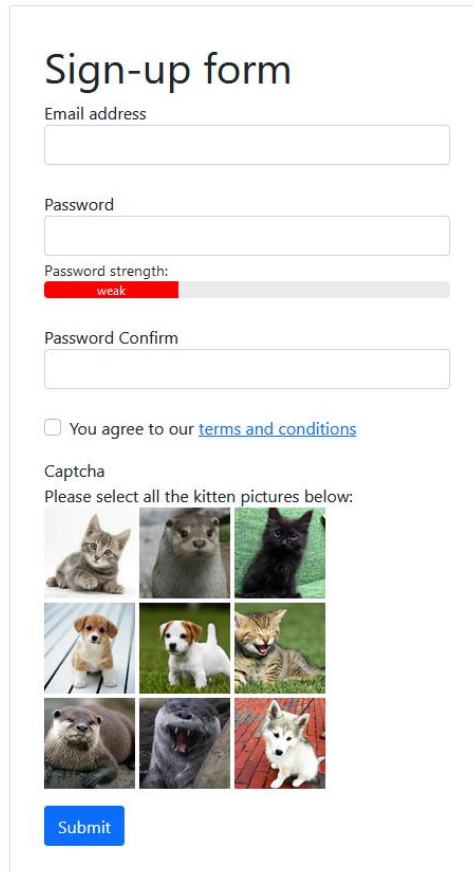
Lorsque cette modale est affichée dans la page, vous devrez voir ceci :

A login modal form with a white background and a black border. It has a title 'Login' at the top left and a close button (X) at the top right. Below the title are two input fields: 'Email address' and 'Password'. At the bottom right, there are two buttons: 'Close' (grey) and 'Login' (blue). A cartoon rabbit is visible on the left side of the modal.

- Lorsque l'utilisateur clique sur le bouton « Login » en haut à droite de la page principale, affichez cette modale.
- La modale doit se fermer lorsque l'utilisateur clique sur le bouton « Close », la croix en haut à droite de la modale, ou bien sur le backdrop derrière la modale.
- Vous n'avez **PAS** besoin de gérer ce formulaire. Rien ne doit se passer lorsque l'on clique sur le bouton « Login »

### C) LA PAGE D'INSCRIPTION

Vous avez également une page d'inscription à faire avec un petit formulaire d'inscription avec quelques vérifications. La page se trouve sur « signup.html » :

A sign-up form titled 'Sign-up form'. It contains the following fields and elements:

- Email address: A text input field.
- Password: A text input field.
- Password strength: A label 'Password strength:' followed by a progress bar showing 'weak' in red.
- Password Confirm: A text input field.
- Agreement: A checkbox labeled 'You agree to our [terms and conditions](#)'.
- Captcha: A label 'Captcha' followed by the instruction 'Please select all the kitten pictures below:'. Below this is a 3x3 grid of nine images. The first three images (top row) are kittens. The other six images (middle and bottom rows) are dogs or other animals.
- Submit: A blue button labeled 'Submit'.

Voici les tâches à faire sur cette page :

- Vérifiez si l'e-mail saisi est une adresse mail valide
- Le mot de passe doit faire au moins 8 caractères
- La confirmation de mot de passe doit correspondre au mot de passe
- La case termes et conditions doit être cochée
- Chaque champ erroné doit être indiqué clairement à côté du champ (il existe une balise HTML préexistante dans la page pour contenir les messages d'erreur)
- Empêchez l'envoi du formulaire si un champ n'est pas bon
- *POUR ALLER PLUS LOIN : Bloquez aussi l'envoi du formulaire si le Captcha n'est pas rempli correctement.*
- *POUR ALLER PLUS LOIN : Implémentez un fonctionnement pour la barre « password strength » juste en dessous du champ mot de passe. Suggestion d'implémentation : Une échelle allant jusqu'à 3 points, où il est possible de « gagner » des points lorsqu'on utilise des variétés de caractères différents. Par exemple, au moins une lettre minuscule donne 1 point, au moins une lettre majuscule donne 1 point, au moins 1 chiffre donne 1 point, et au moins 1 caractère spécial donne 1 point. Pensez à adapter le style avec la couleur de la barre, et le texte à afficher.*