

🧐 D&D: Phase 2 > Viewer de personnages (PHP)

Voici un **cahier des charges complet et structuré** pour la réalisation du visualisateur de personnages Donjons & Dragons en PHP.



Objectif

Développer une **petite application web avec PHP** permettant de visualiser des personnages de type *Donjons & Dragons* (version simplifiée basée sur la 5e édition).

Utilise **Bootstrap** et ses composants pour améliorer le rendu visuel de ton application.



Structure générale de l'application

L'application est constituée de 2 pages PHP avec un **menu de navigation** accessible en permanence.

◆ Pages attendues

Page	URL	Description
Accueil	/	Tableau affichant tous les personnages enregistrés
Fiche personnages	/perso?id=xxx	Fiche affichant les informations sur un personnage



Navigation

Un **menu fixe ou en haut de page** permet d'accéder facilement aux sections:

- Accueil
- Personnages



Liste des personnages

- Affiche tous les personnages de la base dans un **tableau lisible**. Utilise les classes Bootstrap adéquates.

Liste des personnages

Nom	Race	Classe	Niveau	Actions
Arthos	Humain	Guerrier	3	Voir
Elindra	Elfe	Magicien	5	Voir
Krag	Orc	Barbare	2	Voir
Seraphine	Humaine	Clerc	4	Voir
Valdrin	Nain	Voleur	3	Voir
Ygritte	Elfe	Rôdeur	2	Voir
Tharak	Orc	Guerrier	3	Voir
Lyria	Humaine	Sorcière	4	Voir

- Chaque ligne doit contenir :
 - Nom
 - Race
 - Classe
 - Niveau
 - Un bouton de visualisation

Le bouton de visualisation doit contenir un lien qui pointe vers la page de... visualisation (`perso.php`). D'où son nom 😊

◆ Visualisation – Fiche personnage complète

La fiche devrait ressembler à ceci (éventuellement à adapter en fonction des consignes):

BONUS: Affiche la liste des tests réalisés ainsi que leurs résultats

Si tu as réussi à enregistrer les tests de caractéristiques dans la version Python du Créateur de personnages, tu peux les afficher ici.

Comment lire les fiches personnages ?

Voici des fonctions prêtes à l'emploi qui te permettront de lister et lire les fiches personnages créées par ton autre programme.

Idéalement, tu devrais les placer dans leur propre fichier PHP et les inclure dans tes pages à l'aide de `require_once`.

```
/**
1  * Fonction qui charge toutes les fiches personnage d'un dossier donné.
2  *
3  * Elle renvoie un tableau associatif :
4  *   - la clé est le nom du fichier (ex : "actions.json")
5  *   - la valeur est le contenu du fichier transformé en tableau PHP
6  *
```

```

7      * @param string $dossier Chemin vers le dossier (ex : "./data")
8      * @return array Tableau contenant les contenus de chaque fichier JSON
9  */
10     function chargerTousLesPersonnages(string $dossier): array {
11         $resultat = []; // On prépare un tableau vide pour stocker les résultats
12
13         // Vérifie si le dossier existe vraiment
14         if (!is_dir($dossier)) {
15             return []; // Si ce n'est pas un dossier valide, on renvoie un tableau vide
16         }
17
18         // Recherche tous les fichiers qui se terminent par ".json" dans ce dossier
19         $fichiers = glob($dossier . DIRECTORY_SEPARATOR . '*.json');
20
21         // Pour chaque fichier trouvé...
22         foreach ($fichiers as $cheminFichier) {
23             // On récupère le nom du fichier sans le chemin (ex : "mon_fichier.json")
24             $nomFichier = basename($cheminFichier);
25
26             // On utilise la fonction ci-dessous pour lire le contenu du fichier
27             $donnees = chargerFichePersonnage($cheminFichier);
28
29             // Si la lecture du fichier a réussi (pas null), on l'ajoute au tableau
30             if ($donnees !== null) {
31                 $resultat[$nomFichier] = $donnees;
32             }
33         }
34
35         return $resultat; // On renvoie toutes les données chargées
36     }

```

```

/**
1      * Fonction qui lit un fichier JSON précis et transforme son contenu en tableau PHP.
2      *
3      * @param string $chemin Chemin complet vers le fichier (ex : "./data/mon_fichier.json")
4      * @return array|null Un tableau si tout s'est bien passé, sinon null
5  */
6     function chargerFichePersonnage(string $chemin): ?array {
7         // Vérifie que le fichier existe et qu'il est lisible
8         if (!file_exists($chemin) || !is_readable($chemin)) {
9             return null;
10         }
11
12         // Lit tout le contenu du fichier dans une chaîne de caractères
13         $contenu = file_get_contents($chemin);
14
15         // Transforme le texte JSON en tableau PHP (le "true" signifie : tableau associatif)
16         $donnees = json_decode($contenu, true);
17
18         // Vérifie si le décodage s'est bien passé (sinon il y a une erreur)
19         if (json_last_error() !== JSON_ERROR_NONE) {
20             return null; // Erreur de format JSON → on ne garde pas ce fichier
21         }

```

```
22     return $donnees; // On renvoie les données correctement lues
23 }
```

```
1 // ===== EXEMPLES D'UTILISATION (à activer si besoin) =====
2
3 // Charger tous les fichiers JSON d'un dossier "data"
4 $liste = chargerTousLesFichiersJson(__DIR__ . '/data');
5 print_r($liste);
6
7 // Charger un seul fichier JSON
8 $unFichier = chargerFichierJson(__DIR__ . '/data/mon_fichier.json');
9 print_r($unFichier);
```