Explication de fonctionnement de l'aspirateur Web

Décompresser le fichier « aspiweballocine.zip » dans un dossier où votre serveur web peut exécuter son contenu PHP et a les droits d'écriture pour créer dossier et fichier (sur serveur de type wampserver ou autre) . Le script PHP à lancer est : aspirateur.php (qui utilise la bibliothèque PHP simple_html_dom.php du dossier simple HTML DOM à ne pas supprimer et laisser dans le même dossier que le script PHP.

Ce que fait cet aspirateur web personnalisé :

- Il va aspirer, à partir de la racine du site AlloCine pour les séries: https://www.allocine.fr/series-tv/

- Il va parcourir toutes les pages (1121 pages au moment de rédaction de ce document) contenant les titres de séries cliquables sous forme de liens. Pour cela il accède aux pages avec une URL au format :
  https://www.allocine.fr/series-tv/?page=xxx
  (avec xxx : numéro de page de séries)

- Il va, pour chaque série de ces pages, aller voir s'il existe un fichier de critiques des spectateurs qui a le format :
  https://www.allocine.fr/series/ficheserie-xxxxx/critiques/
  (avec xxxxx : code de série)

- Puis il va lire le nombre de pages de commentaires existant pour la critique des spectateurs de cette série (en le lisant sur le bas de page au niveau des liens de parcours des pages) et il va parcourir chacune de ces pages de critique et l'enregistrer sur disque avec un nommage normalisé dans le dossier « rep_aspirateur»:
  critiquesspectxxxx_yyyyy-aaaaa_bbb.html
  xxxx : n° de page de séries parcourue depuis la racine de AlloCine sur les séries.
  yyyy : n° de série enregistré sur disque
  aaaaa : code série AlloCine de la série enregistrée
  bbb : n° de page de commentaires critiques de la série qui a été enregistrée

  Exemple : critiquesspect0002_00022-11042_001.html
  Page des séries n° 0002 / n° de série enregistrée sur disque: 00022 / Code série : 11042 / Page : 001
  C'est donc la 22ème série dont les commentaires ont été enregistrés, de la page 1 des commentaires.

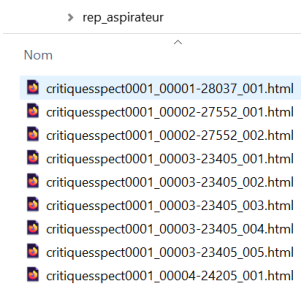
- Il a été conçu pour aspirer avec reprise : si il est interrompu (par le temps maximal d'exécution d'un script PHP, qu'il faut régler au maximum, soit 9999 secondes avec Wampserver si vous avez accès en tant qu'administrateur au serveur = 2h et ¾ d'heures environ) il peut être relancé et va continuer son travail exactement là où il s'était arrêté ; rien n'est perdu ni à refaire. Pour aspirer toutes les critiques de séries du site AlloCine il va falloir des heures, donc relancer plusieurs fois après arrêt au bout de 2h ¾ !

Note : Pour augmenter la limite d'exécution du script à 9999s sans accès admin au serveur on peut utiliser l'instruction suivante du PHP qui permet de régler la durée maximale du script par programmation. Ce code a été inclus.

```
// Met la limite de durée d'excution du script au maximum supporté par Wampserver
// afin que le script s'exécute le plus longtemps possible
set_time_limit(9999);
```

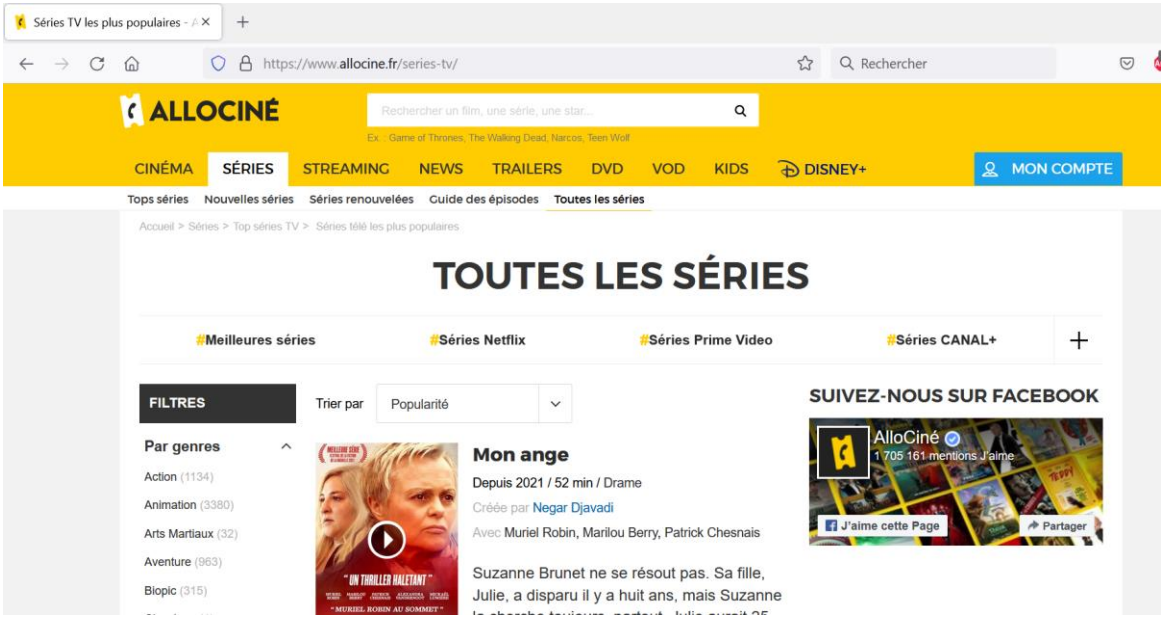
Exemple d'interface produite par le lancement du script PHP de l'aspirateur Web. Il est normal qu'il faille attendre un petit moment pour avoir une mise à jour de l'affichage sur écran. Le navigateur web reçoit les données à afficher du script du PHP mais décide de les afficher à certains moments prévus par un délai interne.

Donc au début s'affiche une page blanche et qui se charge sans cesse sans aucun affichage visible, c'est normal, ne pas s'inquiéter et attendre. Pour voir que le script fonctionne, aller voir le dossier « rep_aspirateur » qui se remplit des fichiers téléchargés.



Explications de construction et compréhension du script PHP de l'aspirateur

On veut aspirer les descriptifs de tous les commentaires pour toutes les séries TV du site allocine. On doit comprendre comment les informations sont organisées et présentées sur le site pour cela. Il faut observer les pages et le code HTML associé pour repérer où sont les informations.



Racine du site de allocine sur les séries TV à l'URL : https://www.allocine.fr/series-tv/

Bas de la page racine des séries TV de allocine : il y a 1121 pages de séries à afficher et la racine affichait en fait la page n°1

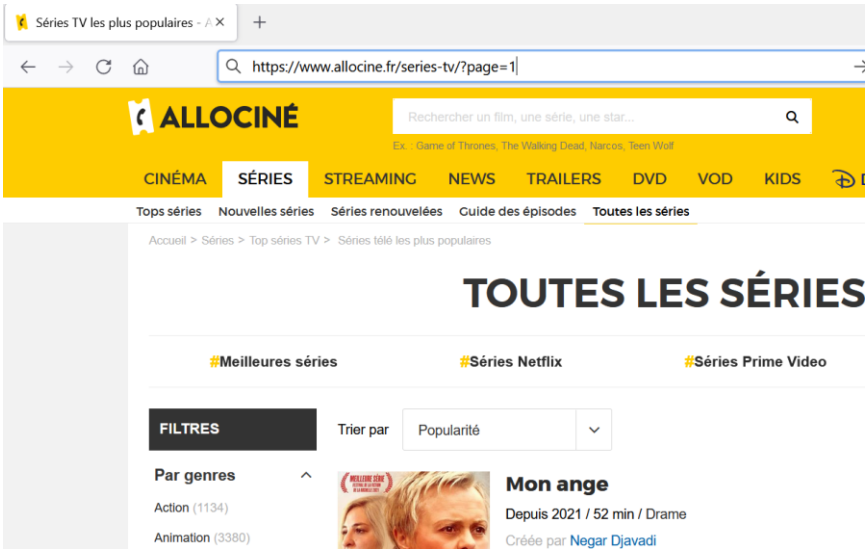
On regarde le code source HTML de la zone qui affiche les numéros de page pour comprendre quels sont les liens pour y accéder (clic-droit, afficher le code source):

```
<a class="rating-title" href="#">Mes amis </a>
<div class="stareval stareval-medium stareval-theme-default"><div class="rating-mdl n00 stareval-stars"><div class="star icon"></div><div class="star ic
</div>
</div>
</div>
<div class="bam-container"><div class="bam-cell bam-rating-user"><span class="user-rating-title">noter :</span><div class="js-user-rating " data-entity-i
</div>
</li>
</ul>
<nav class="pagination cf"><span class="button button-md button-primary-full button-left button-disabled"><i class="icon icon-left icon-arrow-left"></i>
<aside id="gd-col-right" class="gd-col-right flex-col">
```

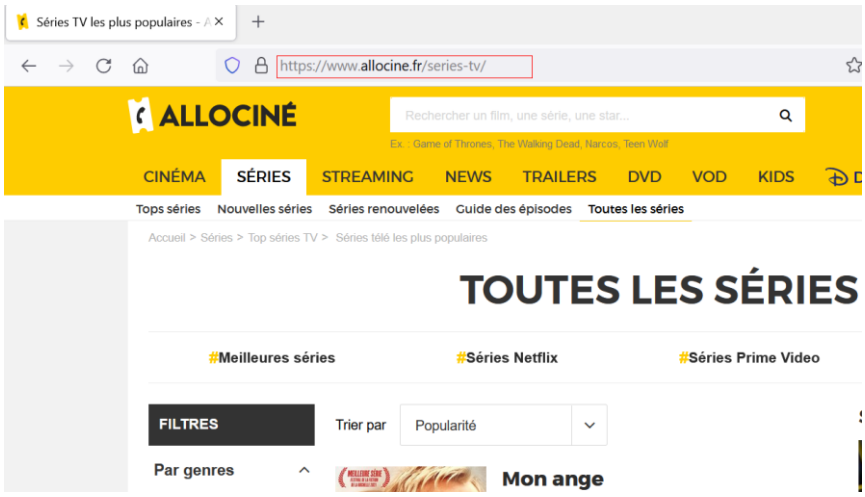
On défile vers la droite le code de la ligne qui contient la balise <nav> qui est la liste des numéros de page cliquable :

```
der"><span class="button button-md item current-item"> 1 </span><a class="button button-md item" href="/series-tv/?page=2">2</a><a class="button button-md item"
<a class="button button-md item" href="/series-tv/?page=3">3</a><a class="button button-md item" href="/series-tv/?page=4">4</a><a class="button button-md
```

On a compris comment accéder aux pages n°i qui listent les séries : ?page=i
Donc l'URL pour avoir accès à la page n°i des séries est : https://www.allocine.fr/series-tv/?page=i



Si on accède à la page numéro 1, on retrouve bien la page d'index



Mais dès qu'on accède à la page, le site supprime (par javascript intégré) l'URL avec le ?page=1 est remplacée dans la barre d'adresse par l'URL racine des séries car c'est la page 1 et que le site décide de supprimé le numéro dans ce cas dans l'URL. Ce n'est pas grave pour nous, mais en effet heureusement qu'on a regardé le code source pour avoir l'URL d'une page car il ne s'affiche pas dans la barre d'adresse pour la page n°1. Il reste bien affiché pour les autres pages.

1) On récupère la liste de toutes les balises « a » du document HTML (les liens) qu'on stocke dans un tableau

Grâce à la bibliothèque « simple_html_dom.php » on crée un objet $html qui contiendra la structure et le contenu d'une page HTML sous la forme d'un objet spécial construit par cette bibliothèque :

```
$html = new simple_html_dom();
```

On parcourt la page n°i (avec l'option de reprise de téléchargement on peut commencer à un numéro donné par =$pagedepart qui provient d'une sauvegarde dans un fichier texte si le téléchargement a été interrompu, sinon on démarre au numéro 1) :

```
// Début réel du téléchargement d'aspiration
for ($i=$pagedepart;$i<=$nbpages_series;$i++){
```

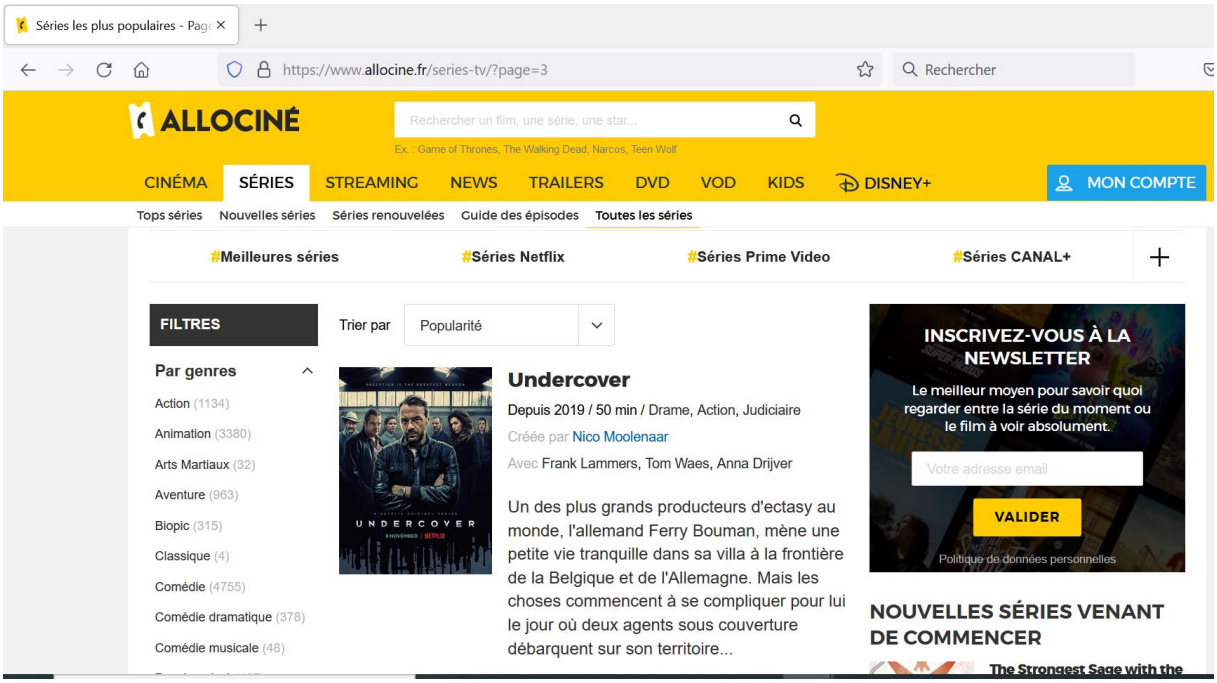
On remplit cet objet à l'aide de la page n°i des séries qu'on souhaite scanner, grâce à la fonction file_get_html :

```
$html=file_get_html($series.'?page='.$i);
```

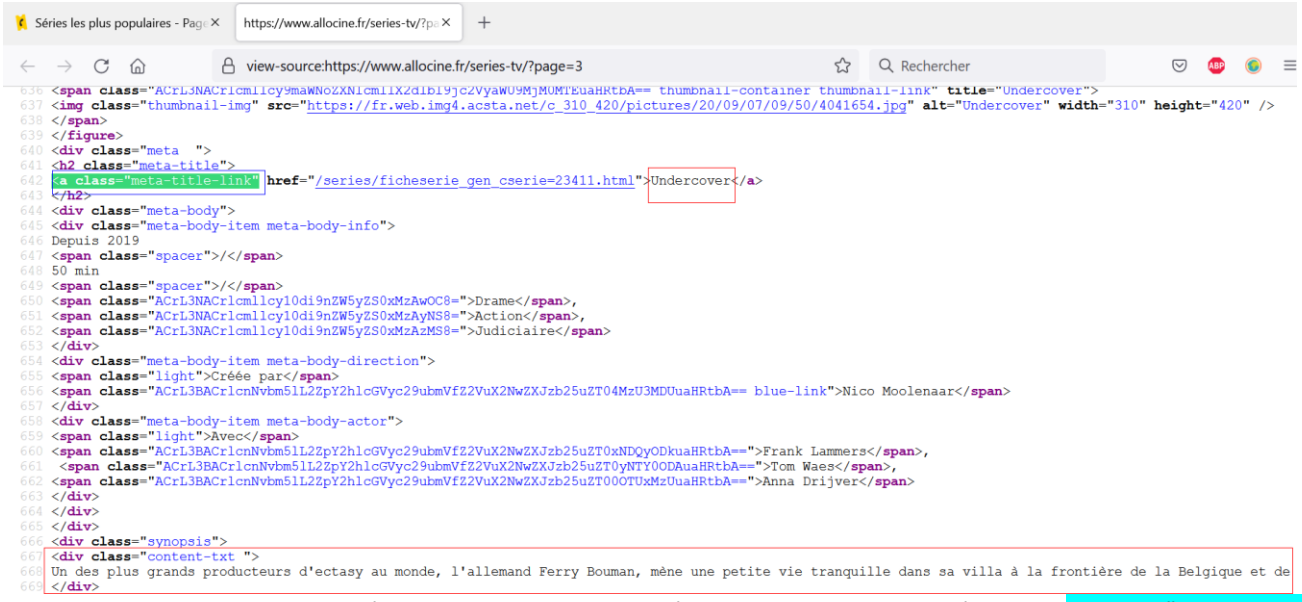
Puis on utilise la méthode find sur l'objet $html qui permet d'aller chercher toutes les balises dont le nom est passé en paramètre, soit « a », qui constitue un tableau d'objets qui est renvoyé, ici on le stocke dans une variable $ret (qui contient un tableau PHP d'objets) :

```
$ret = $html->find("a");
```

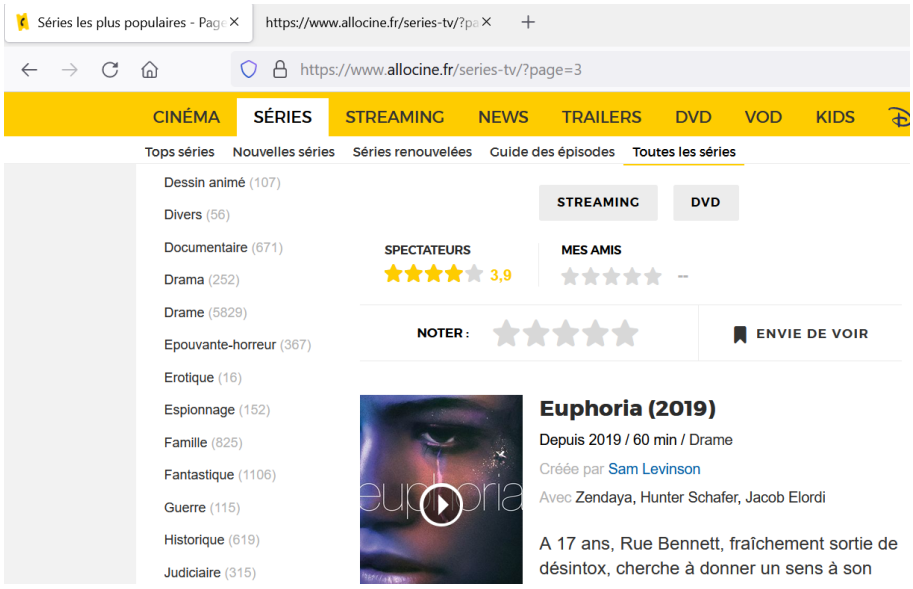
Puis on analyse la façon dont sont placées les zones qui décrivent les séries sur une page donnée :



Exemple : début de la page n°3 de la liste des séries sur le site de allocine – 1ère série affiche : Undercover



On regarde le code source, comment repérer la zone qui contient la série Undercover : la zone débute par <a class="meta-title-link", et le descriptif est ensuite dans un div class= "content-text"



Série suivante de la page n°3 : Euphoria

Encore une fois on voit bien que c'est conforme à notre modèle : la zone débute par <a class="meta-title-link",

2) On va parcourir le tableau des balises « a » précédent et sélectionner seulement celles contenant un attribut « class » et pour lesquelles cet attribut « class » contient la valeur "meta-title-link" afin de trouver une série

On parcourt chaque objet représentant une balise « a » du tableau $ret :

```
foreach($ret as $element) {
```

D'abord une instruction conditionnelle pour voir si l'objet possède un attribut « class » et si cet attribut classe existe si il contient bien le nom de classe recherché "meta-title-link". Chaque attribut de la balise correspond à un champ ou attribut de l'objet correspondant du tableau $ret appelé ici $element :

```
//Si on a trouvé une entrée de série
            if(isSet($element->class) && $element->class=='meta-title-link'){
```

Si on regarde le contenu de l'attribut href de cette balise « a » qui a la bonne classe, elle nous permet d'aller vers la fiche de la série en question. Et on voit que chaque série a un numéro donné par Allocine. Par exemple ci-dessous la série « Undercover » a pour numéro 23411 :

```
642 <a class="meta-title-link" href="/series/ficheserie_gen_cserie=23411.html">Undercover</a>
```

On prend le contenu de l'attribut href et on va le découper selon le caractère « = » qu'il contient afin de récupérer le numéro de série. L'instruction PHP explode produit un tableau contenant pour éléments les chaînes de caractère situées entre chaque caractère « = ». On stocke ce tableau dans la variable $t (« t » comme « tableau ») :

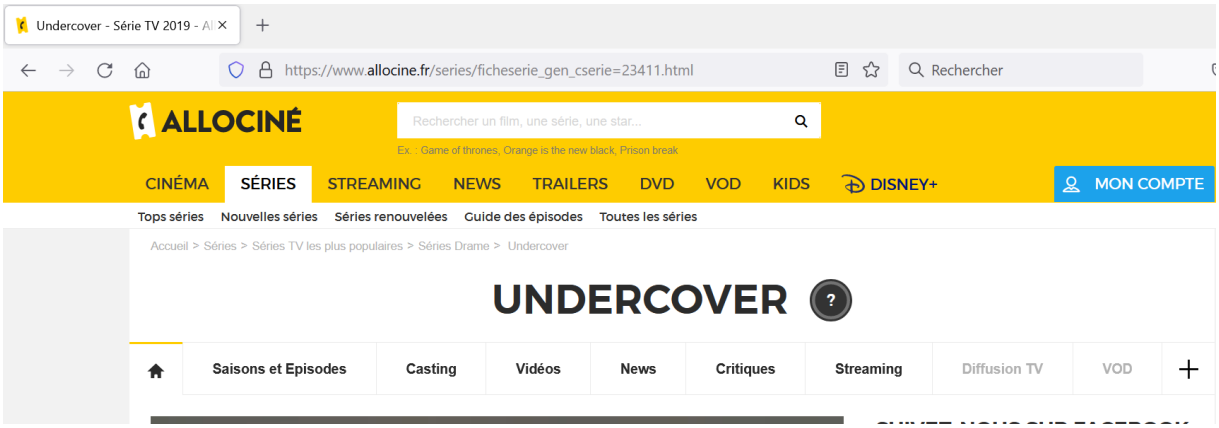
```
$t=explode('=',$element->href);
```

$t contient en fait seulement deux éléments, celui à gauche du « = » en position 0 et celui à droite du « = » en position 1. $t[1] contient donc le numéro de série qui nous intéresse, suivi de « .html ». On veut garder le numéro de série seulement, donc on va découper avec un autre explode selon le « . »

```
$t=explode('.',$t[1]);
```

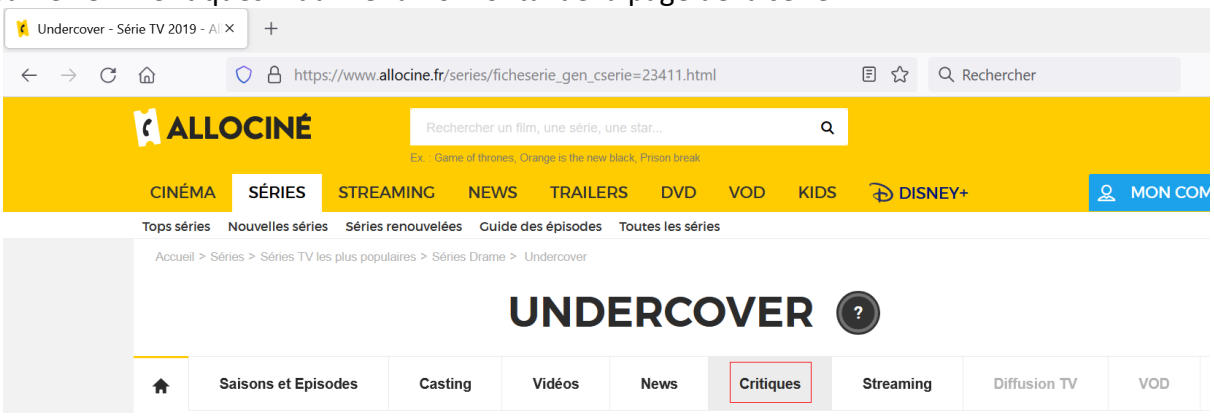
Maintenant $t[0] contient seulement le numéro de série. Vous verrez, on en aura bientôt besoin.

Ce qui va nous intéresser dans un premier temps c'est de consulter la page de la série donnée par $element->href, afin de comprendre où sont les critiques à télécharger.

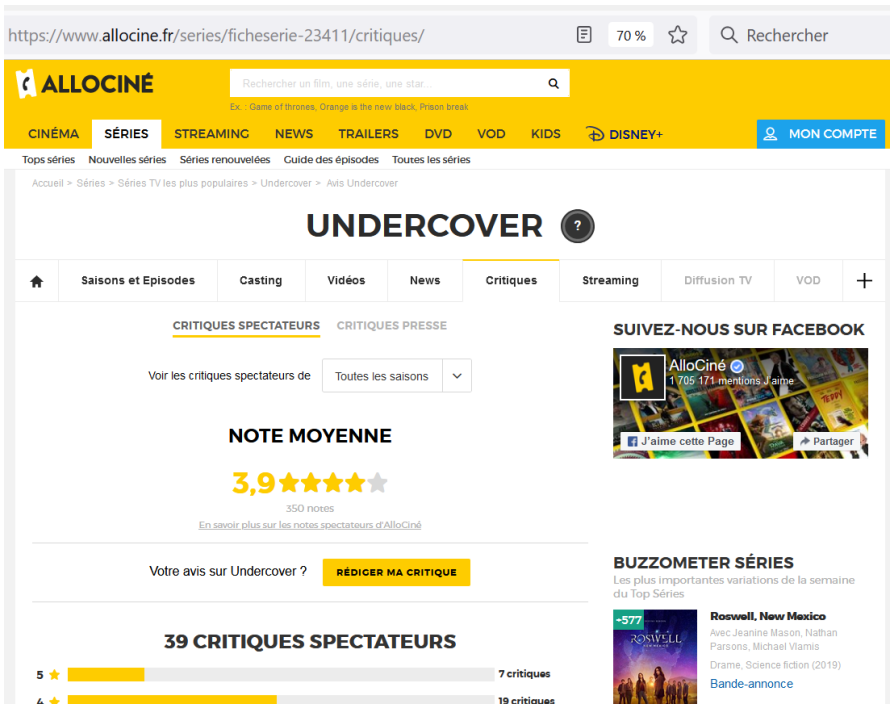


Voilà ce que donne la consultation de la page de la série qui est accessible par l'URL donnée dans https://www.allocine.fr/series/ficheserie_gen_cserie=23411.html

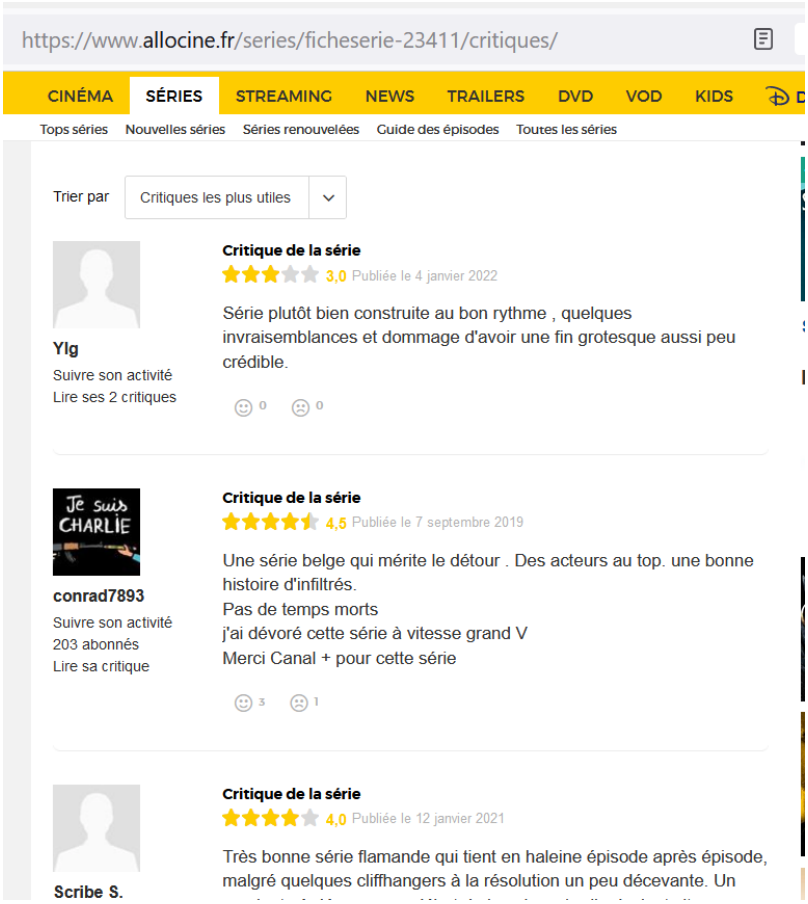
Cliquons sur le lien « Critiques » du menu horizontal de la page de la série :



Cela nous amène à la page des critiques de la série : https://www.allocine.fr/series/ficheserie-23411/critiques/



On défile la page et on peut y lire les critiques (ou commentaires) des spectateurs, c'est bien les données qu'on cherche à télécharger :



Donc voilà une page à télécharger avec notre aspirateur.
On se résume en disant que l'URL qui contient les commentaires de la série est donc :
https://www.allocine.fr/series/ficheserie-23411/critiques/

Et on peut constater que la façon dont allocine a structuré ses URL de commentaires est donc :
https://www.allocine.fr/series/ficheserie-xxxxxx/critiques/

avec xxxxxx : code de la série chez Allocine
Donc si on connaît le code de la série chez Allocine, on peut télécharger la page HTML qui en contient les commentaires. Justement tout à l'heure on a vu que $t[0] contient le code de la série, donc on continue notre téléchargement :

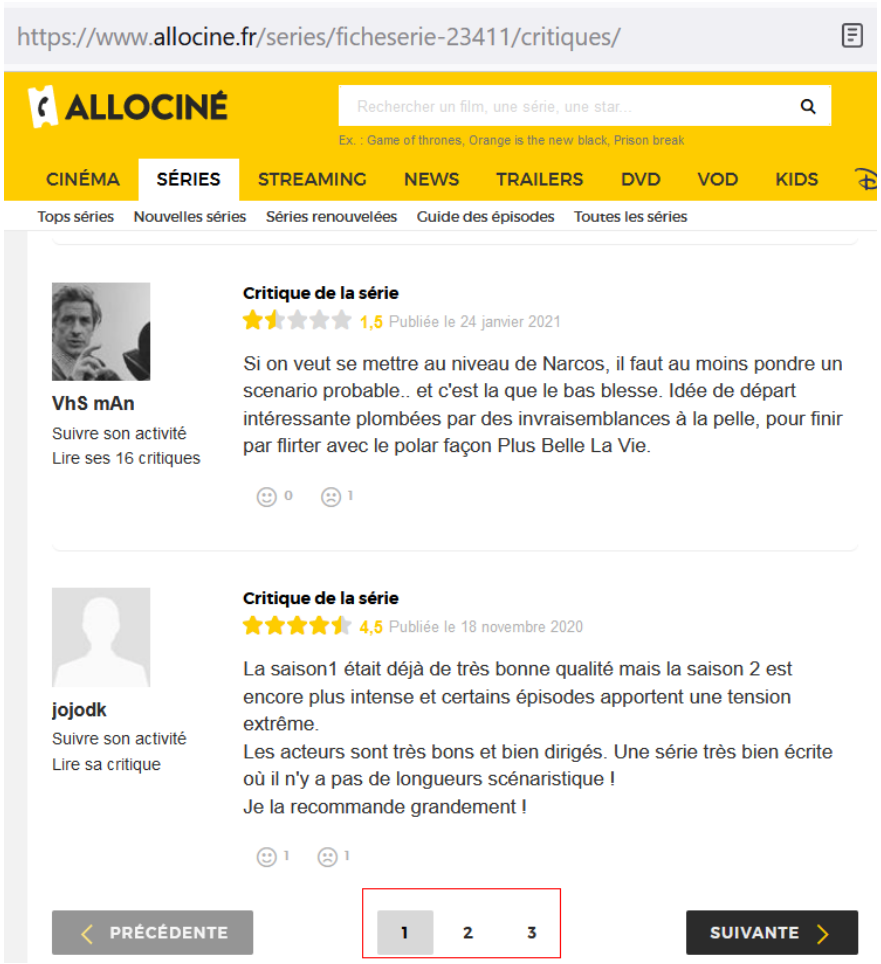
3) Analyse de la page des commentaires de la série trouvée :

Pour que l'utilisateur de l'aspirateur sache ce qu'on fait et où on en est, on l'affiche à l'écran du navigateur :

```
$url_critique_spectateur=$domaine."/series/ficheserie-".$t[0]."/critiques/";
print("<b>Code série: ".$t[0]."</b> - URL critiques: ".$url_critique_spectateur);
```

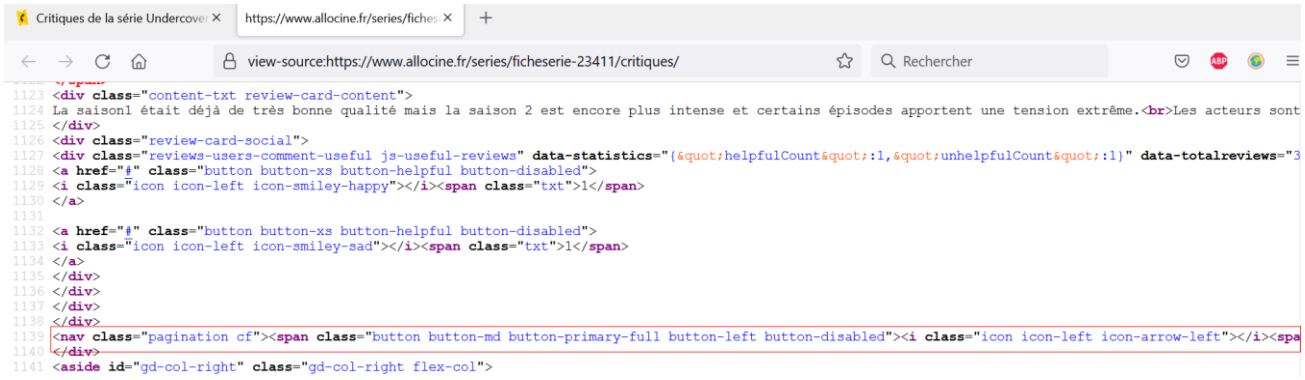
On note que la variable $url_critique_spectateur contient très exactement une adresse du genre
https://www.allocine.fr/series/ficheserie-xxxxxx/critiques/

On pourrait se croire sorti d'affaire, plus qu'à enregistrer la page sur disque, affaire réglée, on passe à la série suivante. Mais non, dommage, c'est plus compliqué que cela car en fait allocine n'affiche qu'un nombre limité de commentaire par page et donc il y a plusieurs pages de commentaires. Constatons-le :



Ici en défilant en bas des commentaires de la série « Undercover » qui nous sert d'exemple, on voit que c'était seulement la page n°1 des commentaires critiques et qu'il y a une page 2 et une page 3

Donc il nous faut accéder à la liste des numéros de commentaires. Regardons le code HTML de nouveau :



Les numéros de page de commentaires cliquables sont dans la balise <nav>, qu'on défile vers la droite :

On constate que les URL pour accéder aux pages des critiques sont du genre :
https://www.allocine.fr/series/ficheserie-xxxxxx/critiques/?page=k
Elles sont lisibles depuis les attributs href des balises <a> qui ont la classe « button button-md item » dans la balise <nav> en question

En fait ce qui nous intéresse seulement, c'est de connaître le nombre de pages de commentaires à télécharger, il suffira alors d'aller systématiquement télécharger les pages avec les URL indiquées ci-dessus.

Or problème, comment savoir quel est le nombre de pages de critiques ?

Prenons un autre exemple avec une série très populaire « La casa de papel » qui a donc de très nombreux commentaires critiques :

On constate ici qu'il y a 76 pages de commentaires. Comment on le sait : le numéro de la dernière page de commentaire est donné comme dernier numéro de la liste des numéros de pages, même si tous les intermédiaires en sont pas listés.

Donc il suffit de parcourir tous les numéros de la liste des liens cliquables de pages de commentaires et de garder le dernier pour savoir quel est le nombre de pages de commentaires.

Toutefois l'exemple d'un grand nombre de pages de commentaires nous montre un nouveau problème généré dans le code de la page Allocine : lorsque le nombre de commentaires est très grand la façon dont sont codés les liens n'est plus la même à partir d'un certain rang. Dans l'exemple qui suit les numéros sont dans des balises <a> au début jusqu'au numéro 2à et ensuite ils sont dans des balises à la place. :

```
">10</a><a class="button button-md item" href="/series/ficheserie-21504/critiques/?page=20">20</a><span class="ACrL3NACrlcmllcy9maWNoZXNlcmllLTIxNTA0L2NyaXRpcXVl

age=20">20</a><span class="ACrL3NACrlcmllcy9maWNoZXNlcmllLTIxNTA0L2NyaXRpcXVlcy8/cGFnZT0zMA== button button-md item">30</span><span class="ACrL3NACrlcmllcy9maWNo
```

Donc il va falloir gérer cette autre difficulté. Comme les numéros sont tous contenus dans des balises <a> de classe « button button-md item » (et seulement dans des balises <a> si il y a peu de pages de commentaires), ou au début dans des balises <a> puis pour la fin dans des balises de classe « button button-md item » (seulement si il y a un grand nombre de pages de commentaires), on va faire la chose suivante :

On va chercher si il y a des balises de classe « button button-md item » auquel cas on va les lire (donc on lit directement les numéros situés à la fin de la liste des numéros).
ATTENTION : pour ces balises span, la classe cherchée est mise à la fin de l'attribut « class » qui contient d'abord une sorte de code crypté aléatoire généré par Allocine.
Si par contre il n'y a pas de balise de classe « button button-md item » alors c'est que les numéros sont contenus seulement dans les balises <a> de classe « button button-md item »

Donc :

On charge la page contenant les commentaires :

```
$htmlcritique=file_get_html($url_critique_spectateur);
```

Puis on va aller chercher toutes les balises « span » de cette page :

```
$retcritique = $htmlcritique->find("span");
```

On parcourt chacune de ces balises span, on regarde si elles ont un attribut « class », et si les 21 derniers caractères de l'attribut contiennent bien le texte « button button-md item », alors on va lire le contenu texte qui est situé entre la balise ouvrante et la balise grâce au champ ou attribut « plaintext » de l'objet PHP $elementcritique qui représente la balise span en question :

```
foreach($retcritique as $elementcritique) {
      if(isSet($elementcritique->class) && substr($elementcritique->class,-21)=='button
   button-md item'){
            if ((int)($elementcritique->plaintext)>$nombre_pages_critiques)
                  $nombre_pages_critiques=(int)($elementcritique->plaintext);
      }
}
```

Le code précédent remet à jour systématiquement le nombre de pages dans la variable $nombre_pages_critiques, en fait il contient à chaque fois le numéro de page contenu entre l'ouvrante et la fermante et lorsqu'on est arrivé au dernier on a donc le numéro le plus grand : c'est ce qu'on cherche.

De plus pour s'assurer qu'il n'y ait pas de problème on remet à jour la variable $nombre_pages_critiques seulement si le numéro de page lu dans le span est plus grand que $nombre_pages_critiques mémorisé précédemment, dès fois qu'il y ait des mauvaises surprises de Allocine sur des fois des balises span et dès fois des balises a selon des cas qu'on n'aurait pas étudiés tous en détail.

Maintenant la même chose avec des balises <a>, et pour ne pas diminuer le numéro de page la plus grande déjà lu par span si on a des et des <a>, on ne met à jour la variable $nombre_pages_critiques seulement si il est plus grand que ce qu'on a déjà lu avec des span

```
//Scan des <a> de classe="button button-md item" (contient le nombre total de pages si peu
élevé)
$retcritique = $htmlcritique->find("a");
foreach($retcritique as $elementcritique) {
      if(isSet($elementcritique->class) && substr($elementcritique->class,-21)=='button
button-md item'){
            if ((int)($elementcritique->plaintext)>$nombre_pages_critiques)
                  $nombre_pages_critiques=(int)($elementcritique->plaintext);
      }
}
```

Ouf, cette difficulté est enfin réglée, on a le nombre de pages de critiques maximal existant pour notre série, et on peut l'afficher à l'écran du navigateur pour l'utilisateur :

```
print("  --> Nb pages: ".$nombre_pages_critiques."<br/>");
```

Remarque :
La même stratégie a été mise en œuvre pour détecter le nombre de séries affichées sur la page d'accueil de Allocine (1121 séries au jour du 09/01/2022). Sauf que comme il y a un grand nombre de séries, on a seulement besoin de faire un scan des et voilà qui explique ce code au début du fichier PHP de l'aspirateur :

```
.tem">1000</span><span class="button">...</span><span class="ACrL3NACrlcmllcy10di8/cGFnZT0xMTIx button button-md item">1121</span></div></nav> </div>
```

```
// Recherche automatique du nombre maximal de séries affichées à la racine de Allocine
$htmlcritique=file_get_html($series);
$retcritique = $htmlcritique->find("span");
$nbpages_series=1;
foreach($retcritique as $elementcritique) {
      if(isSet($elementcritique->class) && substr($elementcritique->class,-21)=='button
button-md item'){
            if ((int)($elementcritique->plaintext)>$nbpages_series)
                  $nbpages_series=(int)($elementcritique->plaintext);
      }
}

print("<h2>Nombre de pages de séries à parcourir détecté: $nbpages_series</h2>");
```

4) Téléchargement des pages de critiques

Ça y est enfin on va pouvoir faire ce qui est important, à savoir télécharger les pages de critiques :
On parcourt tous les numéros de pages de critiques :

```
for ($j=1;$j<=$nombre_pages_critiques;$j++){
```

On charge le contenu HTML de la page n°j en entier d'un coup dans une chaîne de caractère appelée $fichier_critique_page :

```
$fichier_critique_page=file_get_contents($url_critique_spectateur."?page=".$j);
```

Puis on crée le fichier sur disque pour la sauvegarde :

```
$fichier_critique_disque=fopen($chemin_critique_page,'w');
```

Et on écrit le contenu du HTML lu qui est dans la variable $fichier_critique_page vers ce fichier sur disque :

```
fputs($fichier_critique_disque,$fichier_critique_page);
fclose($fichier_critique_disque);
```

Donc le travail est fait !

<u>Au niveau du nom du fichier sauvegardé, voilà ce qui a été fait :</u>

- $i est le numéro de la page de série AlloCine analysée sur laquelle on va aller scanner les séries qui sont dedans.
- $numordre est le numéro d'ordre de la série sauvegardé dans l'ordre de parcours des séries depuis la première série de la première page des séries AlloCine. C'est un numéro croissant allant de 1 en 1 qui n'a rien à voir avec le code de série AlloCine, qui nous sert à une numérotation plus lisible pour nous dans l'ordre chronologique de parcours de téléchargement. Finalement cela nous permet de savoir combien de séries on a chargé.
- $t[0] contient le code de la série sur AlloCine
- $j est le numéro de page de critiques pour la série donnée

On veut que le numéro de page $i ait toujours la même longueur sur 4 chiffres maximum (ce qui permet d'aller jusqu'à 9999 pages ce dont on est loin encore), donc on complète avec des zéros devant le numéro, les zéros pour compléter sont dans la variable $zerosnumpage:

```
      if ($i<10)
            $zerosnumpage='000';
      elseif ($i<100)
            $zerosnumpage='00';
      elseif ($i<1000)
            $zerosnumpage='0';
      else
            $zerosnumpage='';
```

Pareil on a complété avec des zéros pour avoir $numordre sur 5 chiffres fixes, zéros mis dans $zerosnumordre :

```
if ($numordre<10)
      $zerosnumordre='0000';
elseif ($numordre<100)
      $zerosnumordre='000';
elseif ($numordre<1000)
      $zerosnumordre='00';
elseif ($numordre<10000)
      $zerosnumordre='0';
else
      s$zerosnumordre='';
```

Pareil on a complété avec des zéros pour avoir $j sur 3 chiffres fixes, zéros mis dans $zeroscomment :

```
if ($j<10)
	$zeroscomment='00';
elseif ($j<100)
	$zeroscomment='0';
else
	$zeroscomment='';
```

Puis on constitue une norme de nommage du fichier créé sur disque au format critiquesspectxxxx_yyyyy-codeserie_zzz :

```
$chemin_critique_page='./'.$dossier_aspirateur.'/critiquesspect'.$zerosnumpage.$i.'_'.$zerosnumordre.$numordre.'-'.$t[0].'_'.$zeroscomment.$j.'.html';
```

Exemple de nom de fichier sauvegardé provenant de Allocine en 2020:
critiquesspect0022_00376-446_034.html

Provenant de la page de série n°22 : https://www.allocine.fr/series-tv/?page=22
Série n° 376 téléchargée depuis le début de l'aspiration (donc on a traité 375 séries en entier auparavant déjà)
Code allociné de la série : 446 (c'est Allocine qui fixe ses codes de série, donc ça ne dépend pas de nous ce n°)
Page de critiques n°34 de la série chargé

Donc c'est le fichier de critiques n°34 de la série n°376 de code AlloCine 446

Emplacement de stockage :
sous-dossier « rep_aspirateur » qui sera créé automatiquement par le script d'aspirateur PHP si inexistant dans le même dossier que le script de l'aspirateur :

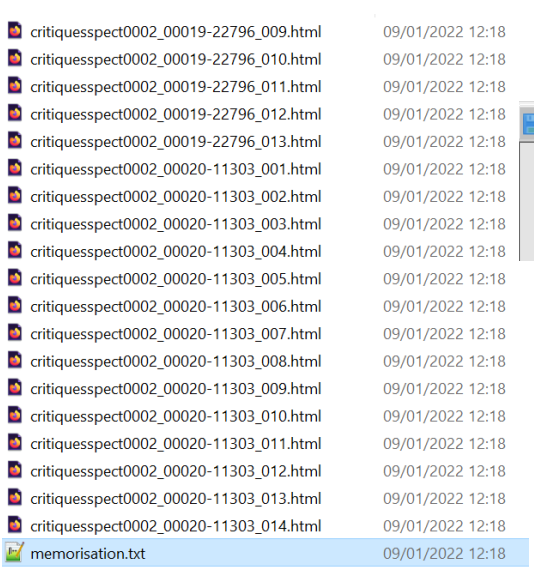
```
//Dossier des captures de critiques faites par l'aspirateur
$dossier_aspirateur='rep_aspirateur';

// Si le dossier "rep_aspirateur" n'existe pas on le crée
if (!is_dir('./'.$dossier_aspirateur))
```

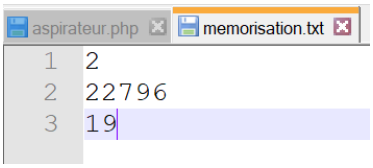
5) La reprise de téléchargement

Si le téléchargement s'est interrompu, pour diverses raisons (coupure internet, script PHP qui arrive à la durée maximale d'exécution autorisée par le serveur, etc) on veut pouvoir poursuivre nos téléchargements sans tout reprendre depuis le début.

Le script prévoit ceci. Un fichier texte est mis à jour à chaque téléchargement terminé d'un fichier de critique de série, indiquant la page de téléchargement, le code Allocine de la dernière série dont toutes les pages de commentaire ont été entièrement téléchargées et le numéro d'ordre de cette série entièrement chargée.



Le fichier texte qui permet la reprise s'intitule « memorisation.txt ». Exemple :



Ici on voit que le dossier « rep_aspirateur » a téléchargé entièrement les critiques de la série n°19 de code AlloCine 22796 et que le téléchargement s'est arrête en cours de la série n°20 de code AlloCine 11303. La mémorisation a retenu ce qui s'est achevé complètement afin de poursuivre avec la suivante si on relance le script : redémarrage du téléchargement de la série de code 11303.

Pour permettre la reprise en début de script PHP de l'aspirateur, on regarde d'abord si le fichier de sauvegarde « memorisation.txt » existe pour redémarrer à partir de là, sinon on commence le téléchargement depuis la page racine des séries TV de Allo cine:

```
// Si le fichier qui mémorise l'en-cours pour reprise de téléchargement existe
// alors on le lit pour recommencer à télécharger là où on s'est arrêté!
if (file_exists($nomfichiermemorisation)){
      $lignes=file($nomfichiermemorisation);
      $pagedepart=(int)($lignes[0]);
      $codeseriedepart=(int)$lignes[1];
      $numordre=(int)$lignes[2];
      print("<h1>Reprise d'aspiration</h1>");
      print("Dernière page de série en cours d'aspiration mémorisée:
".$pagedepart."<br/>");
      print("Dernier code série aspiré mémorisé: ".$codeseriedepart."<br/>");
      print("Dernier n° d'ordre de série aspirée mémorisé: ".$numordre."<br/><br/>");
}
else {
      $pagedepart=1;
      $codeseriedepart=-1;
      $numordre=0;
      print("<h1>Nouvelle aspiration de l'ensemble des séries</h1>");
}
```

Et bien sûr quand on a terminé de télécharger le dernier fichier de critiques d'une série donnée on met à jour ce fichier de reprise (on le crée si il n'existe pas sinon on l'écrase) :

```
//Mémorisation de la position faite pour reprise
$fichier_repriseaspiration=fopen($nomfichiermemorisation,'w');
//Mémorisation de la page des séries en cours d'exploration
fwrite($fichier_repriseaspiration,$i."\n");
//Mémorisation du code série qui vient d'être explorée
fwrite($fichier_repriseaspiration,$t[0]."\n");
//Mémorisation du numéro d'ordre de la série dans la liste des sauvegardes
fwrite($fichier_repriseaspiration,$numordre);
fclose($fichier_repriseaspiration);
```

Comme cette sauvegarde d'information n'est faite qu'après la boucle qui télécharge TOUS les fichiers de critique d'une série donnée, cela permet de bien avoir une information sur une série entièrement chargée et donc c'est la suivante qui a été interrompue et qu'il va falloir redémarrer.

Ici on constate qu'au redémarrage, la série de code 22796 qui était chargée déjà à 12h18 n'a pas été re-téléchargée, mais c'est la suivante qui était interrompue en cours de route qui a été remise en téléchargement à 12h34 : la série de code 11303

6) Interface de sortie écran sur le navigateur et explications :

Nombre de pages de séries à parcourir détecté: 1121

Nouvelle aspiration de l'ensemble des séries

Page des listes de séries: 1

Code série: 28037 - URL critiques: https://www.allocine.fr/series/ficheserie-28037/critiques/ --> Nb pages: 1
https://www.allocine.fr/series/ficheserie-28037/critiques/?page=1
Code série: 27552 - URL critiques: https://www.allocine.fr/series/ficheserie-27552/critiques/ --> Nb pages: 2
https://www.allocine.fr/series/ficheserie-27552/critiques/?page=1
https://www.allocine.fr/series/ficheserie-27552/critiques/?page=2
Code série: 23405 - URL critiques: https://www.allocine.fr/series/ficheserie-23405/critiques/ --> Nb pages: 5
https://www.allocine.fr/series/ficheserie-23405/critiques/?page=1
https://www.allocine.fr/series/ficheserie-23405/critiques/?page=2
https://www.allocine.fr/series/ficheserie-23405/critiques/?page=3
https://www.allocine.fr/series/ficheserie-23405/critiques/?page=4
https://www.allocine.fr/series/ficheserie-23405/critiques/?page=5
Code série: 24205 - URL critiques: https://www.allocine.fr/series/ficheserie-24205/critiques/ --> Nb pages: 5
https://www.allocine.fr/series/ficheserie-24205/critiques/?page=1
https://www.allocine.fr/series/ficheserie-24205/critiques/?page=2
https://www.allocine.fr/series/ficheserie-24205/critiques/?page=3
https://www.allocine.fr/series/ficheserie-24205/critiques/?page=4
https://www.allocine.fr/series/ficheserie-24205/critiques/?page=5
Code série: 11387 - URL critiques: https://www.allocine.fr/series/ficheserie-11387/critiques/ --> Nb pages: 4

Explications des affichages des sorties:

Page des listes de séries: 1

→ indique qu'on est en train de parcourir la page n°1 des séries du site.

Code série: 27552 - URL critiques: https://www.allocine.fr/series/ficheserie-27552/critiques/ --> Nb pages: 2
https://www.allocine.fr/series/ficheserie-27552/critiques/?page=1
https://www.allocine.fr/series/ficheserie-27552/critiques/?page=2

→ indique qu'on est en train de parcourir la série de code AlloCine 27552 des séries et qu'elle comporte deux pages de critiques à télécharger depuis la page des critiques située à l'URL
https://www.allocine.fr/series/ficheserie-27552/critiques/

https://www.allocine.fr/series/ficheserie-27552/critiques/?page=1
https://www.allocine.fr/series/ficheserie-27552/critiques/?page=2

➔ URL des pages de critiques téléchargées pour cette série
On peut vérifier que ces pages ont produit des fichiers sur disque dans le dossier « rep_aspirateur »