

CC PHP : Session 1 - 2h - Sur machine

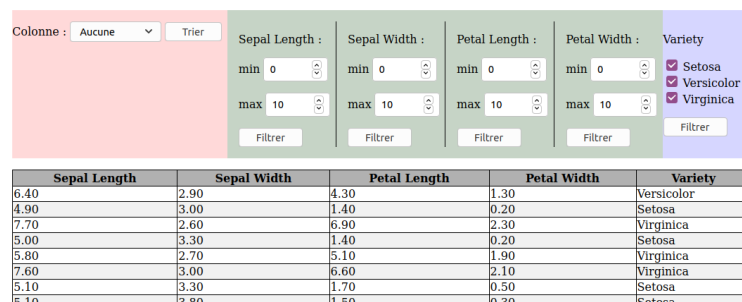
Téléchargez l'archive **cc-20-php-sujet.tgz** de l'espace Moodle. Après extraction, renommez le répertoire obtenu **nom-prenom** où **nom** et **prenom** sont vos nom et prénom écrits en minuscules et sans espaces (par ex. **delafontaine-jean**). A l'issue de l'examen, archivez ce répertoire (`tar czf nom-prenom.tgz nom-prenom`) et déposez l'archive sur Moodle.

Les exercices sont indépendants et peuvent être traités dans l'ordre de votre choix. Pour visualiser ce qui est attendu, consultez ce [démonstrateur](#).

Exercice 1. (Iris de Fisher) Le jeu de données des Iris de Fisher est un jeu de données qui rassemble la longueur et la largeur des sépales et pétales de trois variétés d'Iris. Ce jeu de données a été utilisé pour mettre en relation ces quatre mesures morphologiques de l'Iris avec la taxonomie de l'espèce.

Nous souhaitons créer une page web qui permet de visualiser ce jeu de données, tout en pouvant les trier et les filtrer. Le fichier **iris.csv** présent dans l'annexe contient le jeu de données. La première ligne décrit le type des données de la colonne **f** pour les réels et **s** pour une chaîne de caractères. La deuxième ligne est le nom des données. Les lignes suivantes sont les valeurs des mesures et le type de variété.

Jeu de données des Iris de Fisher



Sepal Length	Sepal Width	Petal Length	Petal Width	Variety
6.40	2.90	4.30	1.30	Versicolor
4.90	3.00	1.40	0.20	Setosa
7.70	2.60	6.90	2.30	Virginica
5.00	3.30	1.40	0.20	Setosa
5.80	2.70	5.10	1.90	Virginica
7.60	3.00	6.60	2.10	Virginica
5.10	3.30	1.70	0.50	Setosa
5.10	3.80	1.50	0.30	Setosa

FIGURE 1 – Vue de la page web au chargement initial.

La figure 1 est une capture du premier chargement de la page où toutes les données sont visibles, non-triées. La page est constituée d'une première partie comportant six formulaires : un formulaire qui gère le tri, quatre formulaires qui permettent de filtrer les quatre mesures, et un formulaire qui permet de filtrer les variétés. La seconde partie est le tableau de données, filtré et trié selon les choix du visiteur.

La première étape va être de réaliser l'affichage des données dans un tableau. La deuxième étape va être de rajouter les formulaires en haut de la page. La dernière étape va être le traitement des données du formulaire.

Pour vous aider à réaliser la page web, le fichier **iris.inc.php** définit plusieurs fonctions. Lisez attentivement la description des fonctions pour pouvoir les utiliser dans les questions suivantes.

1. Modifier le fichier **iris.php** pour ouvrir et lire le fichier **iris.csv**.
2. Modifier la variable `$html_table` du fichier **iris.php** pour y ajouter le contenu du fichier lu dans un tableau. Vous formatez les titres des colonnes des données pour les afficher en tête des colonnes du tableau comme sur la figure 1. Lorsque les données sont numériques, vous les formatez pour qu'elles s'affichent avec deux décimales.
3. Modifier la variable `$html_form_tri` du fichier **iris.php** pour produire l'affichage du formulaire de tri. Il est composé d'un élément "select" qui permet de sélectionner l'ensemble des colonnes des données, et d'un bouton "submit".
4. Modifier la variable `$html_form_filtre_float` du fichier **iris.php** pour produire l'affichage des formulaires de filtre des mesures. Chaque formulaire est composé d'un paragraphe indiquant le nom de la mesure, d'un autre paragraphe comportant un élément "number" pour filtrer la valeur minimale, un troisième paragraphe permettant de filtrer la valeur maximale, et enfin un bouton "submit".

Jeu de données des Iris de Fisher

Colonne : Sepal Length ▼ Trier

Sepal Length :
min 0
max 10
Filtrer

Sepal Width :
min 0
max 10
Filtrer

Petal Length :
min 0
max 10
Filtrer

Petal Width :
min 0.5
max 1
Filtrer

Variety
☒ Setosa
☒ Versicolor
☒ Virginica
Filtrer

Sepal Length	Sepal Width	Petal Length	Petal Width	Variety
4.90	2.40	3.30	1.00	Versicolor
5.00	2.00	3.50	1.00	Versicolor
5.00	3.50	1.60	0.60	Setosa
5.00	2.30	3.30	1.00	Versicolor
5.10	3.30	1.70	0.50	Setosa
5.50	2.40	3.70	1.00	Versicolor
5.70	2.60	3.50	1.00	Versicolor
5.80	2.70	4.10	1.00	Versicolor
6.00	2.20	4.00	1.00	Versicolor

FIGURE 2 – Vue de la page web avec les iris dont le “Petal Width” est entre 0.5 et 1, triés par “Sepal Length”.

- 5. Modifier la variable \$html_form_filtre_str du fichier **iris.php** pour produire l’affichage du formulaire de filtre des variétés. Il est composé d’un paragraphe indiquant le nom de la colonne représentant les variétés, d’un second paragraphe listant les variétés disponible dans le fichier, et d’un bouton "submit".
- 6. Modifier le fichier **iris.php** pour ajouter le traitement des données des différents formulaires. Pour se souvenir des données précédemment envoyées, vous utiliserez soit les cookies soit les sessions.
- 7. En fonction des paramètres choisis par l’utilisateur, filtrer et trier le tableau de données avant de l’afficher.

Voici une proposition d’organisation de votre code dans le fichier **iris.php** :

```
#1 Traitement des données des formulaires
#2 Lecture du fichier iris.csv
#3 Génération des formulaires
#4 Filtrage, tri et génération du tableau de visualisation
```

Exercice 2. (Expressions régulières)

Nous voulons développer un moteur de recherche d’articles scientifiques. Ce moteur utilise le fichier **articles.txt** comme base de données, avec trois types de requêtes possibles :

- 1. Recherche du titre d’articles qui contient une chaîne quelconque.
- 2. Recherche du titre d’articles qui commencent par une chaîne quelconque.
- 3. Recherche du titre d’articles qui finissent par une chaîne quelconque.

Recherche d'articles

Resultat

[Audemard2008] Experimenting small changes in CDCL Algorithms
[Audemard2008] SAT graph-based representation - A new perspective
[Audemard2009] Predicting Learnt Clauses Quality in Modern SAT Solvers
[Audemard2012] Refining restarts strategies for SAT and UNSAT
[Audemard2015] Glucose and Syrup in the SAT Race 2015
[Audemard2020] SAT Heritage

Recherche d'articles

Resultat

[Baillieux2002] Une bibliotheque C++ pour le codage CNF de contraintes arithmetiques
[Steinke2015] PBLib - A C++ Toolkit for Encoding PB Constraints into CNF

Recherche du type “Au début”

Recherche du type “contient”

FIGURE 3 – Exemples de fonctionnements du moteur de recherche.

Pour traiter les caractères spéciaux (réservés aux expressions régulières) dans le motif de recherche, utilisez la fonction preg_replace(). Le moteur considérera seulement les parenthèses (()), les crochets [] et le signe plus (+).

Le fichier **articles.txt** liste la référence de l'article (nom de l'auteur et année de publication) entre crochets, puis le nom du fichier PDF qui stocke l'article. Les noms de fichiers sont toujours nommés de la même manière que le titre de l'article. Il est donc possible de réaliser une recherche sur le titre de l'article en faisant la recherche sur le nom du fichier, en prenant soin de ne pas prendre en compte l'extension `.pdf`.

1. Compléter le fichier **regex.php** avec le code PHP nécessaire pour traiter les données venant du formulaire, générer le motif de recherche, traiter le fichier et afficher les résultats correspondants comme illustré en figure 3.

Exercice 3. (Templates et variables globales) Dans cet exercice toutes les superglobales `SERVER` devront être déclarées en constante dans le fichier **main.inc.php**. La page **main.php** doit afficher le contenu de la Figure 4 ci-dessous et chaque page (**header.php**, **body.php** et **footer.php**) doit être capable d'afficher sa partie indépendamment.

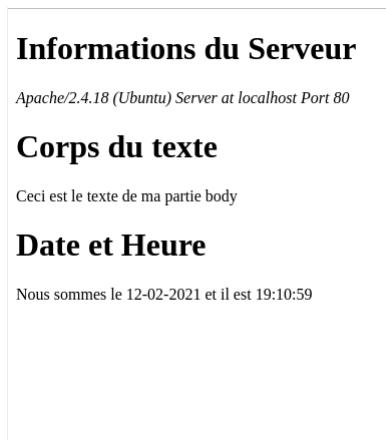


FIGURE 4 – Rendu de l'exercice 3

1. **Header** : en utilisant la fonction `writeContentHeader()` (présente dans **main.inc.php**), **header.php** doit être capable d'afficher les informations du Server.
2. **Body** : Modifier la fonction `writeContentBody()` (présente dans **main.inc.php**) afin que la variable `$contentBody` soit affichée entre les balises `<p></p>`.
3. **Footer** : en utilisant la fonction `writeContentFooter()` (présente dans **main.inc.php**), **footer.php** doit être capable d'afficher la date et l'heure de la requête.
4. Pour finir, compléter le fichier **main.php** afin qu'il puisse afficher l'ensemble du document (header, body et footer).

Exercice 4. (Fonctions sur les tableaux) Pour chaque question le résultat attendu est présenté par la Figure 5. Pour ce faire, vous complétez le fichier **callbacks.php**.

1. En utilisant une fonction de rappel (*call-back*) afficher pour chaque élément du tableau `$characters`, n fois le caractère (où n correspond à un nombre de répétition contenu dans le tableau `$repeat`). Pour `$characters = ['p', 'h', 'p']` et `$repeat = [1, 2, 5]` le résultat attendu est `phhppppp`.
2. En utilisant une fonction de rappel (*call-back*) trier le tableau `$mots` par ordre décroissant du nombre de lettre de chaque mot. Attention aux clefs (voir exemple de la Figure 5).
3. Sans utiliser de fonction de rappel (*call-back*) afficher, pour chaque élément du tableau `$a`, sa clef et sa valeur en alternant les deux affichages suivants : "clef-valeur" et "valeur-clef" (voir exemple de la Figure 5).

Question 1 :

phhppppp

Question 2 :

Avant triage :

```
/home/php_dev/Mes_projets_web/dev-web2/php/cc1/callbackEtTableau/exercice.php:20:
array (size=6)
  0 => string 'université' (length=11)
  1 => string 'LERIA' (length=5)
  2 => string 'licence' (length=7)
  3 => string 'informatique' (length=12)
  4 => string 'CC' (length=2)
  5 => string 'php' (length=3)
```

Après triage :

```
/home/php_dev/Mes_projets_web/dev-web2/php/cc1/callbackEtTableau/exercice.php:24:
array (size=6)
  3 => string 'informatique' (length=12)
  0 => string 'université' (length=11)
  2 => string 'licence' (length=7)
  1 => string 'LERIA' (length=5)
  5 => string 'php' (length=3)
  4 => string 'CC' (length=2)
```

Question 3 :

```
/home/php_dev/Mes_projets_web/dev-web2/php/cc1/callbackEtTableau/exercice.php:34:
array (size=4)
  'Jean' => string 'Pierre' (length=6)
  'Ama' => string 'Castor' (length=6)
  'Bon' => string 'Courage' (length=7)
  'Onyme' => string 'Anne' (length=4)
```

Jean-Pierre
Castor-Ama
Bon-Courage
Anne-Onyme

FIGURE 5 – Rendu de l'exercice 4