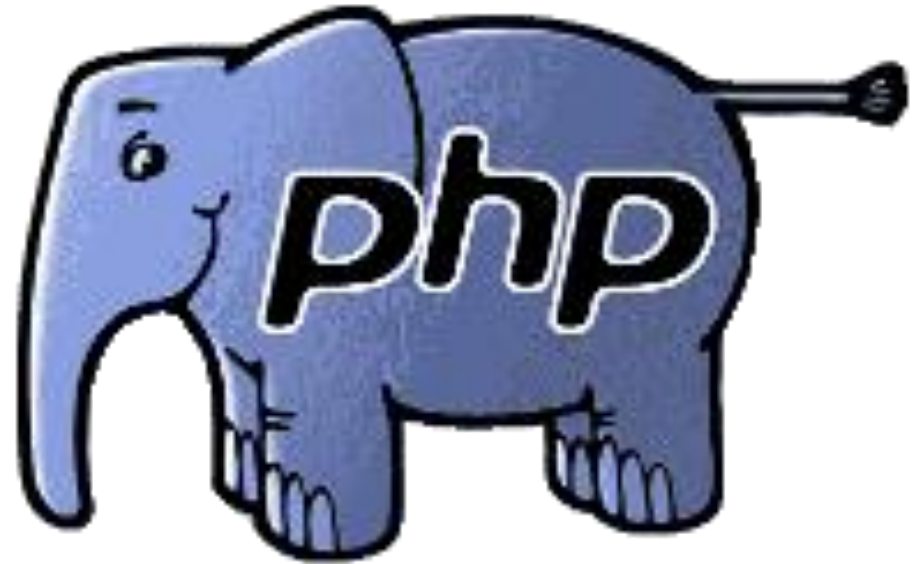


## PHP : Fichier / csv POO



### Connaissances pré-requises

- PHP : Condition - Boucle - Inclusion - Fonction


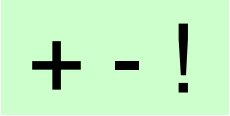
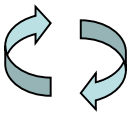
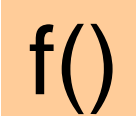






## Fichier / csv

- [Création](#)
- [Options](#)
- [Exemple](#)
- [Fichier csv](#)

## POO

- [Concept](#)
- [Création](#)
- [Utilisation](#)
- [Exemple](#)

# PHP - Fonctionnalités dans ce document

- \* Manipulation de **variables** (locale, globale, constantes). 
- \* **Opérations** avec les variables, caractères, dates. 
- \* Les **boucles** (if, else, ternaire, switch, for, while). 
- \* La factorisation et les **fonctions**. 
- \* Les **tableaux** de variables. 
- \* Le passage de **paramètres** (URL) à une autre page web, formulaires (get, post).
- \* Manipulation des **répertoires et fichiers** sur le serveur.
- \* Fabrication de code 
- \* Gestion des **sessions** et **cookies** (sur le poste utilisateur). 
- \* PHP et Base de donnée. 
- \* **Paramétrage**  , **Sécurité** (formulaires). 
- \* Programmation Orientée Objets (**POO**)

# Création de fichier dynamiquement

**Les Fichiers** peuvent permettre de stocker des données en vue d'une réutilisation ultérieure avec une page web ou un logiciel autre (Exemple : fichiers.csv avec MS-Excel). Le contenu de ces fichiers est accessible par une lecture ligne à ligne.

## Exemple

```
$res=  
fopen(  
"files/recette-fichier.txt",  
"a+" );
```

```
fputs(  
$res,  
$fichier);
```

```
$r = opendir($repertoire);  
readdir($r)
```

```
$contenuentableau =  
file($repertoire.$lecturenomfichier);
```

## Explications

Variable de connexion au fichier  
Instruction PHP pour **ouvrir un fichier**  
Emplacement, Nom et Extension du fichier à ouvrir  
Si ce nom de fichier n'existe pas => création du fichier

Instruction PHP **pour écrire dans un fichier**  
Variable de connexion au fichier  
Contenu à écrire dans le fichier associé à la connexion

Instruction PHP pour **ouvrir un dossier**  
Instruction PHP pour lire le nom d'un fichier d'un dossier

Nom de la variable pour stocker le contenu du fichier lu.  
Instruction PHP pour **découper les lignes** du contenu d'un fichier et les insérer dans un tableau PHP

# Création de fichier dynamiquement (suite)

Option d'ouverture	Description
<b>r</b>	Ouvre le fichier en lecture seule. Cela signifie que vous pourrez <b>seulement lire</b> le fichier.
<b>r+</b>	Ouverture en lecture et écriture Si ce nom de fichier n'existe pas => création du fichier <b>curseur positionné au début</b> (attention remplacement 1er caractères)
<b>a</b>	Ouvre le fichier en écriture seule. Mais il y a un avantage : si le fichier n'existe pas, il est automatiquement créé.
<b>a+</b>	Ouverture en <b>lecture et écriture</b> Si ce nom de fichier n'existe pas => création du fichier <b>curseur positionné à la fin</b>

# Création de fichier (Instructions complémentaires)

```
fgetc($monfichier); // permet de lire le 1er caractère d'un fichier  
fgets($monfichier); // permet de lire la première ligne d'un fichier  
fread($monfichier, 27); // permet de lire les 27 premiers caractères d'un fichier  
$tab = file($monfichier); // mise en tableau du fichier (1 ligne fichier = 1 ligne tableau)  
filesize($monfichier); // compte le nombre de caractères du fichier  
count($tab); // compter le nb de ligne du tableau  
fseek($monfichier, 0); // repositionne le curseur au premier caractère du tableau  
rename($anciennomfichier, $nouveaunom); // renomme un fichier  
unlink($nomfichierasupprimer); // supprime un fichier  
copy($source , $dest) // copier le contenu d'un un fichier dans un autre  
$section = file_get_contents('./people.txt', FALSE, NULL, 20, 14);  
// Lit 14 caractères, à partir du 21ème caractère
```

# Création de fichier dynamiquement

## Les Fichiers - Exemple de code

```
1 <?php /* fichier cnam/nfa017/fichier.php -- 20170503-PBO */
2
3 /* entête HTML ===== */
4 <<<include("inc/header.inc.php"); headerinclude("fichier");
5
6 /* variables ===== */
7 <<<$fichiername="recette-fichier.txt";
8 <<<$fichierrep="files";
9 <<<$fichiertext = "Carottes rapées"; /* texte que je souhaite écrire dans mon fichier */
10
11 /* CREATION du fichier et de son contenu ===== */
12 <<<$res=fopen($fichierrep."/".$fichiername,"a"); // crée le fichier s'il n'existe pas
13 <<< // sinon ouvre le fichier en mode écriture
14 <<< // positionne le curseur d'écriture à la fin du fichier
15 <<< fputs($res, $fichiertext); // insère les données dans le fichier
16 <<< fclose($res); // fermeture du fichier
17
18 /* fin du HTML ===== */>>
19 <p>Dans le répertoire "<?php echo $fichierrep; ?>"<br/>
20 Je créer le fichier "<?php echo $fichiername; ?>" et <br/>
21 j'y mets "<?php echo $fichiertext; ?>"</p>
22
23 <p><a href="index.php">retour vers index.php</a></p>
24
25 </body>
26 </html>
```

```
index.php x connexion.inc.php x fichier.inc.php x |
1 <!-- fichier cnam/nfa017/inc/fichier.inc.php -- 20170503-PBO -->
2
3 <strong>Fichier : </strong><?php
4 <<<$lecturenomfichier="";
5 <<<$repertoire="files";
6 <<<$repertoire="./".$repertoire."/";
7 <<<$r = opendir($repertoire); // ouverture du répertoire inc
8 <<<while($lecturenomfichier = readdir($r)) {
9 <<<    if($lecturenomfichier == "recette-fichier.txt") {
10 <<<        $contenuentableau = file($repertoire.$lecturenomfichier); // lignes du fichier => array
11 <<<        foreach ($contenuentableau as $don) {echo $don.'<br/>';} // affichage lignes du array
12 <<<        unlink($repertoire.$lecturenomfichier); // suppression du fichier
13 <<<    }
14 <<<} ?>
```

## Les Fichiers (extension. csv)

permettent de

- Transférer des données d'une base de donnée vers un tableur.
- Transférer des données d'un tableur vers une Base de Données.

Ces fichiers au format texte sont relativement léger (Cela dépend du volume de données qu'ils contiennent).

## Ils peuvent être fabriqués par :

- Une base de donnée,
- Une page web,
- Un tableur,
- Un autre logiciel.



## Propriétés des Fichiers CSV

**csv** : extension reconnue par les principaux tableurs (excel, openOffice, ...)

**format** : texte

**charset** : convertir fichier en ANSI (pour éviter les problèmes d'accents)

Instruction PHP pour convertir :

```
$texte_ansi = iconv('UTF-8', 'Windows-1252', $texte_utf8);
```

**séparateur** : point virgule : " ; "

**Encadrement des données** : guillemets doubles : " ... .. " ;  
(sauf pour les nombres qui n'ont pas besoin de guillemets)

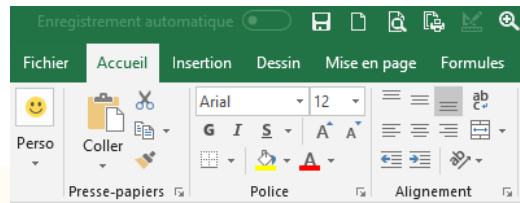
**Retour à la ligne** (inséré avec instruction PHP) :

```
$texte_avec_retour_ligne = $texte_sans_retour_ligne."\\n"
```

**Possibilité d'ajouter des formules**

# Fichiers CSV : Exemple de code

```
creation-fichier.php x
1 <?php /* .VARIABLES .*/
2 ..$texte_pour_fichier .= "recette"; "temps de préparation" . "\n"; ;
3 ?>
4
5 <!DOCTYPE HTML><html lang="fr"><!-- fichier.cnam/nfa016/conf20161123/index.php.20170000-PBO.-->
6 <head><meta charset="utf-8" ./>
7 .....<title>Création fichier Recettes</title>
8 .....<link href="css/style.css" rel="stylesheet" type="text/css" ./>
9 .....<meta name="Description" content="DESCRIPTION" ./>
10 .....<meta name="viewport" content="width=device-width" ./>
11 </head>
12 <html><body>
13 <h1>Création fichier Recettes</h1><?php
14
15 /* .MENU .===== .*/
16 ..include "inc/menu.inc.php";
17
18 /* .RECUPERATION .INFOS .BDD .===== .*/
19 ..include "inc/connexion.inc.php";
20 ..$requete = "SELECT recette_id, recette_nom, recette_preparationtemps FROM recette";
21 /**/ ..echo $requete;
22 ..$resultat = $con->query($requete);
23
24 /* .AFFICHAGE .INFOS .DE .LA .BDD .===== .*/
25 ..while($donnees = $resultat->fetch()) { .?>
26 .....<p><?php echo $donnees['recette_nom'] . " | " . $donnees['recette_preparationtemps'];
27 .....if ($donnees['recette_preparationtemps'] == "") {$donnees['recette_preparationtemps'] = "?"; .}
28 .....$texte_pour_fichier .= " " . $donnees['recette_nom'] . " ";
29 .....$donnees['recette_preparationtemps'] . " min" . "\n"; .?>
30 .....</p><?php
31 .....}
32 /**/ ..echo "texte stocké : " . $texte_pour_fichier . "<br/>";
33 ..unlink("files/recette-fichier.csv"); /* .supprimer .un .fichier .*/
34 ..$texte_pour_fichier_ansi = iconv('UTF-8', 'Windows-1252', $texte_pour_fichier); /* .convertir .au .charset .lisible .par .excel .*/
35 ..$res = fopen("files/recette-fichier.csv", "a+"); /* .ouverture .fichier .+ .création .si .inexistent .*/
36 ..fputs($res, $texte_pour_fichier_ansi); /* .insertion .du .texte .dans .le .fichier .*/
37 /**/ ..echo "nb caractères du fichier créé : " . filesize("files/recette-fichier.csv") . "<br/>"; .?>
38 <footer> Copyright P. Bouquet pour cnam - Mentions légales </footer>
39 <!-- .Fermeture .du .fichier .HTML .===== .-->
40 </body></html>
```



	A	B
1	recette	temps de préparation
2	Gâteaux aux 2 chocolats	30 min
3	Pizza à l italienne	? min
4	Spaghettis bolognaises	9 min
5	entrée avocat crevette	20 min
6	Tarte au chocolat	12 min



# Programmation Orientée Objet (POO)

Attribut  
de la classe  
(Variable)

**Classe : recette**

duree :

afficherduree()  
modifduree()

Méthode de  
la classe  
(Fonctions)

objets créés à  
partir de la classe  
recette

**Objet : gateau1**  
duree : 20 min

**Objet : gateau2**  
duree : 1h

**Remarque :** Ne pas confondre

**POO** : Programmation Orientée Objet et

**PDO** : Php Data Object, pour se connecter à une BDD

## Création d'un objet

### Création de la classe

```
<?php class maclasse {          attributs          méthodes          }
```

### Attributs d'une classe

Ex : private **\$mvariable** = 20 ; /\* déclaration avec valeur par défaut \*/

### Méthodes d'une classe

```
public function mafonction1() { return $this-> mvariable ; }  
public function mafonction2($saisie) {$this-> mvariable = $saisie; }
```

### Remarques

**public** : accès depuis n'importe où

**private** : accès uniquement qu'à partir intérieur de la classe

**\$this** : représente l'objet en cours d'utilisation

**return** : permet d'utiliser le résultat de la méthode en dehors de celle-ci.

Lors de l'utilisation des **attributs** dans les méthodes, le signe \$ disparaît.

## Utilisation d'un objet et de ses propriétés

### Création d'un objet (à partir d'une classe existante)

```
$monobjet1 = new maclasse ;           /* déclaration avec valeur par défaut */  
$monobjet2 = new maclasse ;           /* déclaration d'un autre objet */
```

### Utilisation des Méthodes d'un objet

```
$monobjet1 -> mafonction1() ; /* utilisation de la fonction d'un objet */  
$monobjet1 -> mafonction2(35) ; /* utilisation d'une autre fonction avec  
paramètre  
pour le même objet */  
$monobjet2 -> mafonction1() ; /* utilisation d'une fonction pour un autre objet */
```

# Programmation Orientée Objet - Exemple

## Exemple de code

```
recette-poo.php x
1  <?php /* fichier nfa017/recette-poo.php 20170531-PBO */
2
3  /* Creation d'une classe de ses attributs et ses methodes */
4  class recette{ /* definition de la classe recette ===== */
5      /* Attributs */ private $duree=20;
6      ...
7      /* Methodes */ public function afficherduree(){return $this->duree;}
8      public function modifierduree($dureesaisie){$this->duree = $dureesaisie;}
9  }
10
11 /* entete HTML ===== */?>
12 <!DOCTYPE html><html>
13 <head><meta charset='UTF-8' /><title>NFA017 -- Recette POO</title>
14 <meta name="description" content="Exercice PHP POO -- Classe recette"/>
15 </head>
16 <body><h1>NFA017 -- Recette POO</h1>
17 <section><?php
18
19 /* Utilisation de la classe recette ===== */
20 $gateaul = new recette; ?>
21 <p>Durée gateaul initial : <?php
22     echo $gateaul->afficherduree(); ?></p><?php /* affichage : 20 */
23     $gateaul->modifierduree(30); ?>
24 <p>Durée gateaul modifié : <?php
25     echo $gateaul->afficherduree(); ?></p><?php /* affichage : 30 */
26     $gateau2 = new recette; /* Création d'un 2e objet de type gateau */ ?>
27 <p>Durée gateau2 : <?php
28     echo $gateau2->afficherduree(); ?></p><?php /* affichage : 20 */ ?>
29 <p>Durée totale d'utilisation : <?php
30     echo ($gateaul->afficherduree()+$gateau2->afficherduree()); ?></p>
31 <?php /* affichage : 50 */ ?>
32 </section>
33 </body>
34 </html>
35
```