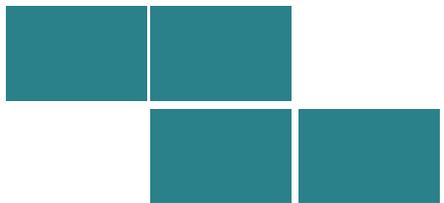


Chapitre 8

Dates et calendriers



I. Introduction

- ❑ Pour déterminer la validité des cookies, stocker dans une base de données des informations de date de commande ou calculer un délai, on doit gérer le temps.
- ❑ Les informaticiens ont défini une date d'origine arbitraire, correspondant au **01/01/1970 00 h 00 m 00 s**. À partir de cette date, le temps est compté en **secondes**. Ce nombre de secondes est nommé **timestamp**, ou instant UNIX.

Problème: **Timestamp négatif**

II. Dates et heures

Les fonctions de gestion des dates et heures sont très utiles. Le PHP en fournit plusieurs. En voici les plus intéressantes:

- 1) int **time**(): Renvoie l'heure actuelle en nombre de secondes depuis l'époque Unix (1er janvier 1970 00:00:00). Elle est calculé **coté serveur** (cette valeur n'est pas affichée aux visiteurs du site).

Exemple : Ecrire un script qui permet de calculer :

- Le timestamp a cet instant précis.
- Le timestamp dans 23 jours.
- Le timestamp il y a 12 jours.
- Le nombre d'heure depuis 1/1/1970.

II. Dates et heures

Correction : `<meta charset="utf-8" />`

`<?php`

`echo "A cet instant le timestamp est :",time(),"
";`

`$tp=time()+(23*24*3600);`

`echo "Dans 23 jours le timestamp sera: $tp", "
";`

`$tps=time()-(12*24*3600);`

`echo "Il ya 12 jours le timestamp était :$tps", "
";`

`$tpss=round(time()/3600);`

`echo "Le nombre d'heure depuis 1/1/1970=$tpss", "
";?>`

II. Dates et heures

2) array `getdate`(int \$timestamp): Retourne un tableau contenant toutes les informations de date.

Exemple 1: Dans cet exemple, on suppose que la date est le **mardi 07 décembre 2021 à 15h 38min 53s**

```
<?php
$stamp=time ();
Echo "<pre>" ;
print_r (getdate ($stamp) );
Echo "</pre>" ;
?>
```

Résultat de l'exécution :

```
Array ( [seconds] => 53 [minutes] => 38 [hours] => 15 [mday] => 7 [wday] => 2 [mon] => 12
[year] => 2021 [yday] => 340 [weekday] => Tuesday [month] => December [0] =>
1638887933 )
```

Remarque :

Si le **timestamp** est omis ou null, il vaut par défaut la valeur de la fonction **time()**.

II. Dates et heures

Clé	Description
wday	Le jour de la semaine sous forme d'entier de 0 (dimanche) à 6 (samedi)
weekday	Le jour de la semaine sous forme de chaîne (en anglais)
mday	Le jour du mois sous forme d'entier de 0 à 31
mon	Le mois sous forme d'entier de 1 à 12
month	Le mois sous forme de chaîne (en anglais)
year	L'année en entier sur 4 chiffres
hours	L'heure de 0 à 23
minutes	Les minutes de 0 à 59
seconds	Les secondes de 0 à 59
yday	Le jour de l'année de 1 à 366
0	Le timestamp correspondant à la date

II. Dates et heures

Exemple 2:

Ecrire un script qui permet d'afficher le message suivant : Nous somme le 01/11/2024

Correction: Le code PHP est:

```
<?php
$ladate=getdate ();
$jour=$ladate ["mday"];
$mois=$ladate ["mon"];
$annee=$ladate ["year"];
echo "Nous sommes le $jour/$mois/$annee";
?>
```

II. Dates et heures

- 3) string **date**(string \$format, int \$timestamp): Retourne une chaîne de caractères date/heure selon le format spécifié et représentant la date et l'heure système de la machine du serveur web. Les formats possible sont rassemblés dans le tableau suivant:

Caractère de définition	Définition et résultat affiché
y	L'année en deux chiffres (05 pour 2005)
Y	L'année en quatre chiffres (2005)
L	Affiche 1 si l'année est bissextile et 0 sinon.
m	Le mois en deux chiffres de 01 à 12
n	Le mois en un ou deux chiffres de 1 à 12
M	Le mois en trois lettres (en anglais)
F	Le mois en toutes lettres (en anglais)

II. Dates et heures

t	Le nombre de jours du mois de 28 à 31
d	Le jour du mois en deux chiffres de 01 à 31
j	Le jour du mois en un chiffre de 1 à 31
D	Le jour de la semaine en trois lettres (en anglais)
l (petit L)	Le jour de la semaine en toutes lettres (en anglais)
w	Le jour de la semaine codé de 0 pour dimanche à 6 pour samedi
S	Affiche le suffixe anglais « th » ou « nd » après les chiffres du jour.
z	Le jour de l'année de 0 à 366
g	Les heures de 1 à 12 (avec AM et PM)
h	Les heures de 01 à 12 (avec AM et PM)
G	Les heures de 0 à 23
H	Les heures sur deux chiffres de 00 à 23

II. Dates et heures

a	Ajoute « am » pour le matin ou « pm » pour l'après-midi.
A	Ajoute « AM » pour le matin ou « PM » pour l'après-midi.
i	Les minutes en deux chiffres de 00 à 59
s	Les secondes en deux chiffres de 00 à 59
U	Affiche le timestamp UNIX.
Z	Donne le décalage horaire par rapport au temps GMT ou UTC en seconde, de -43 200 à 43 200.
T	Affiche la ville significative du fuseau horaire, par exemple « Paris, Madrid ».
I	Affiche 0 pendant l'heure d'hiver et 1 pendant l'heure d'été.
r	Affiche la date complète au format RFC 822, par exemple: « Sun, 13 Apr 2003 22:34:46 +0200 ».
B	Heure Internet Swatch : invention de la société Swatch selon laquelle 24 heures sont divisées en 1 000 éléments nommés « beats ». Par exemple, midi vaut 500 beats.

II. Dates et heures

Exemple: écrire le script qui permet d'avoir ces résultats.

Si on utilise le format:....., La date affichée serait : Tuesday 7 December 2021

Et avec le format:....., La date affichée serait :7/12/2021

Et avec le format:....., la date affichée serait :Tue 2021-12-07

Et avec le format:,la date affichée serait :Tue 07/12/21 19:23:02

Correction :

```
<?php
```

```
echo "Si on utilise le format: l j F Y, La date affichée serait :".date("l j F Y"),  
"<br/>";
```

```
echo " Et avec le format: j/n/Y, La date affichée serait :".date("j/n/Y"), "<br/>";
```

```
echo " Et avec le format: D Y-m-d la date affichée serait :".date("D Y-m-d"),  
"<br/>";
```

```
echo " Et avec le format: D d/m/y H:i:s la date affichée serait :".date("D d/m/y  
H:i:s");
```

```
?>
```

II. Dates et heures

- 4) booléen `checkdate`(\$mois, \$jour, \$année): Cette fonction vérifie la validité d'une date.

Exemple :

```
<?php
var_dump (checkdate (12, 12, -600) ) ;
var_dump (checkdate (12, 31, 2015) ) ;
var_dump (checkdate (12, 45, 2021) ) ;
?>
```

Résultat de l'exécution :

```
boolean false
boolean true
boolean false
```

II. Dates et heures

- 5) int **mktime**(\$heure, \$minute, \$seconde, \$mois, \$jour, \$année) : Retourne l'instant UNIX (**timestamp**) d'une date spécifiée (après 1er janvier 1970).
- 6) int **gmmktime** (\$heure, \$minute, \$seconde, \$mois, \$jour, \$année) : Retourne le **timestamp** correspondant à la date GMT.

Exemple :

Ecrire un script qui permet d'obtenir le résultat suivant :

En instant UNIX, le timestamp de 1/11/1990 est : 632012400

En GMT, le timestamp de 1/11/1990 est : 632016000

Correction :

```
<?php
```

```
$tp=mktime(0,0,0,11,1,1990);
```

```
echo "En instant UNIX, le timestamp de 1/11/1990 est : ".$tp.
```

```
"<br>";
```

```
$t=gmmktime(0,0,0,11,1,1990);
```

```
echo "En GMT, le timestamp de 1/11/1990 est : ".$t. "<br>";
```

```
?>
```

III. Les fonctions de calendrier

- 1) `int easter_date(int année)`: Retourne le timestamp du jour de Pâques de l'année passée en paramètre. Cette fonction permet de fabriquer un calendrier complet pour une année donnée.

Exemple :

```
<?php
```

```
echo "Paques 2022 sera : ", date("d F Y", easter_date(2022));
```

```
?>
```

Résultat de l'exécution :

Paques 2022 sera : 17 April 2022

III. Les fonctions de calendrier

- 2) `int gregoriantojd(int mois, int jour, int année)`: Retourne le nombre de jours du calendrier Julien.
- 3) `jdtogregorian(int nbr_jour_julien)`: convertit un nombre de jours juliens en date grégorienne "mois/jour/année".

Exemple :

```
<?php
$tpjd=gregoriantojd(3,15,2014);
echo $tpjd. "<br/>";
echo jdtogregorian($tpjd);
?>
```

Résultat de l'exécution :

```
2456732
3/15/2014
```

III. Les fonctions de calendrier

5) `int cal_days_in_month(int $calendar, int $month, int $year)`: Retourne le nombre de jours dans un mois pour une année et un calendrier spécifiés .

Exemple :

```
<meta charset="utf-8" />
<?php
$j_r=cal_days_in_month(CAL_GREGORIAN,2,1998);
echo "Le nombre de jours dans le mois de février de l'année 1998
est: $j_r jours", "<br/>";
$j_rr=cal_days_in_month(CAL_GREGORIAN,5,2008);
echo "Le nombre de jours dans le mois de mai de l'année 2008
est: $j_rr jours";
?>
```

Résultat de l'exécution :

Le nombre de jours dans le mois de février de l'année 1998 est: 28 jours

Le nombre de jours dans le mois de mai de l'année 2008 est: 31 jours

III. Les fonctions de calendrier

- 6) string **jdtofrench**(int \$nombredejoursdecalendrierjulien) : Convertit le nombre de jours du calendrier julien en date du calendrier républicaine française.
- 7) int **frenchtojd**(int \$mois, int \$jours, int \$annee): Convertit une date du calendrier français republicain en nombre de jours du calendrier julien.

Exemple :

```
<?php  
$jr=frenchtojd(1,7,14);  
echo $jr. "<br/>";  
echo jdtofrench($jr);  
?>
```

Résultat de l'exécution :

```
2380594  
1/7/14
```

TP : Dates et calendriers

Exercice 1:

Ecrire un script qui permet d'avoir ce résultat, puis déterminez la date et l'heure de l'exécution de ce script.

Résultat de l'exécution :

A cet instant le timestamp est :.....

Dans 23 jours le timestamp sera:

Il ya 12 jours le timestamp était :.....

Le nombre d'heure depuis 1/1/1970=.....

Le nombre de jours depuis 1/1/1970=.....

La date et l'heure d'exécution de ce script est :

TP : Dates et calendriers

Correction:

```
<meta charset="utf-8" />
<?php
echo "A cet instant le timestamp est :", time(), "<br/>";
$tp=time()+(23*24*3600);
echo "Dans 23 jours le timestamp sera: $tp", "<br/>";
$tps=time()-(12*24*3600);
echo "Il ya 12 jours le timestamp était :$tps", "<br/>";
$tpss=round(time()/3600);
echo "Le nombre d'heure depuis 1/1/1970=$tpss", "<br/>";
$tpsss=round(time()/(3600*24));
echo "Le nombre de jours depuis 1/1/1970=$tpsss", "<br/>";
$dat=date("d/m/Y H:i:s", time());
echo "La date et l'heure d'exécution de ce script est : $dat", "<br/>";
?>
```

Exercice 2:

Calculez votre âge à l'instant en cours à la seconde près, puis à l'heure près, ensuite à jour près.

Correction:

```
<?php
$timp=time()-mktime(0,0,0,11,1,1990);
echo "Mon age est en seconde près est: $timp", "<br/>";
$t=$timp/3600;
$h=round($t);
echo "Le nombre des heures vécu est : $h", "<br/>";
$tt=$t/24;
$j=round($tt);
echo "Le nombre des jours vécu est : $j", "<br/>";
?>
```

TP : Dates et calendriers

Exercice 3:

Vérifiez si la date du 29 février 1962 a existé

Correction:

```
<?php  
  
var_dump (checkdate (02, 29, 1962) ) ;  
  
?>
```

Résultat de l'exécution :

boolean false

TP : Dates et calendriers

Exercice 5:

Affichez toutes les années bissextiles comprises entre 2005 et 2052, ainsi que le nombre de ces années.

Correction:

```
<?php
$nb=0;
for ($i=2005;$i<=2025;$i++) {
if (date("L",mktime(0,0,0,1,1,$i))==1) {
echo "$i est une année bissextile", "<br/>"; $nb++; }
}
echo "Le nombre des années bissextiles entre 2005 et 2025 est :
$nb", "<br/>";
?>
```

Exercice 5 (Suite):

Résultat de l'exécution :

2008 est une année bissextile

2012 est une année bissextile

2016 est une année bissextile

2020 est une année bissextile

2024 est une année bissextile

Le nombre des années bissextiles entre 2005 et 2025 est : 5

Exercice 6:

Déterminez quel jour de la semaine seront tous les premier Mai des années comprises entre 2005 et 2010. Si le jour est un samedi ou un dimanche, affichez le message « Désolé ! ». Si le jour est un vendredi ou un lundi, affichez « Week-end prolongé ! ».

Correction:

```
<?php
for ($i=2005;$i<=2010;$i++) {
$tmp=mktime(0,0,0,5,1,$i);
setlocale(LC_TIME,"fr");
$d=strftime("%A",$tmp);
if ($d=="samedi" || $d=="dimanche") {
echo "Désolé" , "<br/>"; }
if ($d=="lundi" || $d=="vendredi") {
echo "Weekend prolongé" , "<br/>"; } }
?>
```

Exercice 6 (suite):

Résultat de l'exécution :

Désolé

Weekend prolongé

Weekend prolongé

Désolé

Exercice 7:

L'Ascension est le quarantième jour après Pâques (Pâques compris dans les 40 jours).

Calculez les dates de l'Ascension pour les années 2005 à 2010.

TP : Dates et calendriers

Correction:

```
<?php
for ($i=2005;$i<=2010;$i++) {
$asc=easter_date($i)+(40*24*3600);
echo "L'Ascension de l'année ".$i." était le : ". date("d F
Y",$asc),"<br/>"; }
?>
```

Résultat de l'exécution :

L'Ascension de l'année 2005 était le : 06 May 2005
L'Ascension de l'année 2006 était le : 26 May 2006
L'Ascension de l'année 2007 était le : 18 May 2007
L'Ascension de l'année 2008 était le : 02 May 2008
L'Ascension de l'année 2009 était le : 22 May 2009
L'Ascension de l'année 2010 était le : 14 May 2010