

## **Entorns de Desenvolupament**



### 16. Fitxers: Introducció

## 16. Fitxers

Quan necessitem crear, guardar, llegir o escriure arxius en la creació de llocs web, el maneig d'aquests es converteix en una prioritat. Per a això, anem a veure una sèrie de funcions i alguns exemples per familiaritzar-nos amb el maneig d'arxius a nivell bàsic en php.

### fopen

La funció fopen() serveix per obrir fitxers. La seua sintaxi general:

```
<?php

    $fp=fopen(fitxer,modeObertura);

?>
```

On **\$fp** és el descriptor o identificador del fitxer obert que necessitarem més tard.

El paràmetre **fitxer** pot ser un arxiu de text, un arxiu amb extensió .php, o fins i tot la URL d'una pàgina web (per exemple http://www.paginaweb.com/). "Obrir" un web amb **fopen()** pot ser molt útil quan volem comprovar si el web existeix, o si està actiu.

- Si **fitxer** comença amb "http://", s'obre una connexió cap al web especificat. Sempre ha posar-se una barra (/) al final.
- Si **fitxer** comença amb "ftp://", s'obre una connexió ftp al servidor especificat.
- Si **fitxer** no comença amb cap de les coses anteriorment dites, s'obre una connexió "directa" amb l'arxiu especificat. Si no existeix l'arxiu o direcció especificades, es retorna un error.

A l'hora d'obrir un arxiu, cal saber de quina manera volem obrir-lo. Podem obrir-lo per a escriptura i lectura, però de diferents formes:

Mode	Observacions
'r'	Obertura per només lectura; col·loca el punter al fitxer al principi del fitxer.
'r+'	Obertura per a lectura i escriptura; col·loca el punter al fitxer al principi del fitxer.
'w'	Obertura per només escriptura; col·loca el punter al fitxer al principi del fitxer i trunca el fitxer a longitud zero. Si el fitxer no existeix s'intenta crear.
'w+'	Obertura per a lectura i escriptura; col·loca el punter al fitxer al principi del fitxer i trunca el fitxer a longitud zero. Si el fitxer no existeix s'intenta crear.
'a'	Obertura per només escriptura; col·loca el punter del fitxer al final del mateix. Si el fitxer no existeix, s'intenta crear. ( <b>append</b> )
'a+'	Obertura per a lectura i escriptura; col·loca el punter del fitxer al final del mateix. Si el fitxer no existeix, s'intenta crear.
'x'	Creació i obertura per només escriptura; col·loca el punter del fitxer al principi del mateix. Si el fitxer ja existeix, la crida a fopen() fallarà tornant FALSE i generant un error. Si el fitxer no existeix s'intenta crear.
'x+'	Creació i obertura per a lectura i escriptura; d'una altra manera té el mateix comportament que 'x'.
'c'	Obrir el fitxer per només escriptura. Si el fitxer no existeix, es crea. Si existeix no és truncat (a diferència de 'w'), ni la crida a aquesta funció falla (com en el cas amb 'x'). El punter a fitxer es posiciona en el principi del fitxer. Això pot ser útil si es vol obtenir un bloqueig assistit abans d'intentar modificar el fitxer, ja que en usar 'w' es podria truncar el fitxer abans d'haver obtingut el bloqueig.
'c+'	Obrir el fitxer per a lectura i escriptura; d'una altra manera té el mateix comportament que 'c'.

**Exemple:**

```
<?php // Exemple
    $fp=fopen("./files/fitxer.txt","r");
    $fp=fopen("./files/fitxer2.txt","w");
    $fp=fopen("http://www.iessimarro.org/fitxer.txt","a+");
    $fp=fopen("ftp://ftp.iessimarro.org/fitxer.txt","w");
?>
```

Ara bé, amb aquesta simple instrucció només tindrem l'arxiu obert per llegir, escriure, o llegir i escriure. Però ara haurem de fer ús de les altres instruccions per escriure o recuperar les dades que estímem oportunes.

**fclose**

La funció **fclose()** tanca un arxiu obert.

```
<?php
    fclose (identificadorDelFitxer)
?>
```

Aquesta funció ha de ser utilitzada després d'obrir i manipular l'arxiu com veurem en els següents exemples.

**fgets i feof**

La funció **fgets()** recupera el contingut d'una línia d'un arxiu. La seua sintaxi és:

```
<?php
    fgets(descriptorDelFitxer);
?>
```

Exemple:

```
<?php
// Llegim la primera línia de fitxer.txt
$fp=fopen("fitxer.txt","r");
// fitxer.txt han d'estar a la mateixa carpeta que el fitxer php
$linia=fgets($fp);
// fitxer.txt és un arxiu de text normal.
fclose($fp);
?>
```

Ara bé, amb això només llegirem la primera línia del fitxer de text, si volguérem llegir línia a línia fins al final necessitaríem l'ús d'un **bucle while**, per **exemple**.

```
<?php
//Llegirem línia a línia fitxer.txt fins a la fi (feof ($fp))
$fp=fopen("fitxer.txt", "r");
while (!feof($fp)) {
    $linia=fgets($fp);
    echo $linia."<br/>";
}
fclose($fp);
?>
```

Això ens mostrarà el contingut del fitxer de text línia a línia. Comprovem que hem inclòs una nova funció anomenada **feof()**, que vol dir **file end of file** o "marca de final de fitxer"). La sintaxi general per aquesta funció és:

**feof (identificadorDelArxiu)**

La funció **feof()** ens retorna:

- **true** quan hem arribat al **final d'arxiu**
- **false** si **no hem arribat encara**.

La condició `while (!feof($fp))` podríem haver-la escrit també de la següent manera:

```
while (feof($fp)==false)
```

Aquestes sintaxis són equivalents, es pot utilitzar qualsevol de les dues.

Finalment, una vegada hem acabat d'operar amb el fitxer escrivim la instrucció:

```
fclose ($fp)
```

per tancar la connexió, cosa que **hem de fer sempre**.

## **fwrite / fputs**

La funció `fputs()` és totalment equivalent a la funció `fwrite()` i **escriu una línia en un arxiu**. La seua sintaxi general és:

```
<?php

    fputs(descriptorDelArxiu,cadena);

?>
```

**Exemple:**

```
<?php // Escrivim una primera línia a fitxerEscriure.txt
    $fp=fopen("fitxerEscriure.txt","w");
    fputs($fp,"Prova d'escriptura" );
    fclose($fp);

?>
```

En realitzar l'obertura del fitxer en mode w, **si el fitxer no existeix, serà creat**.

D'igual manera, una vegada hem de acabar d'operar amb el fitxer escrivim la instrucció

```
fclose ($fp)
```

per tancar la connexió, cosa que hem de fer sempre.

Si ara obrim el fitxer amb algun editor de textos com Notepad, Bloc de notes o qualsevol altre podrem veure el seu contingut.

És recomanable, quan manipulem fitxers, **no continuar amb l'execució de les instruccions de tractament del contingut si hi ha hagut un error a l'hora d'obrir-lo**, per això es pot ficar en una instrucció condicional. L'exemple d'abans, seria:

```
<?php // Comprovem primer si es pot obrir el fitxer.  
    if($fp=fopen("fitxerEscriure.txt","w")){  
        fwrite($fp,"Prova d'escriptura" );  
        fclose($fp);  
    } else {  
        echo "error obrint el fitxer";  
    }  
?>
```

En sistemes de fitxers que diferencien entre fitxers binari i de text, com Windows, per a escriure'l amb **fwrite()**, el fitxer s'ha d'obrir incloent el valor **"b"** en el mode d'obertura.

```
$fp=fopen("fitxer.txt", "wb")
```

Per defecte, **fwrite** / **fputs** **no insereixen un final de línia a les cadenes** que introduïm, si volem fer-ho, podem afegir el símbol **'\n'** al final de la cadena o concatenar el valor **PHP\_EOL** al final d'allò que volem escriure al fitxer.

Per exemple:

```
<?php
    if($fp=fopen("fitxerEscriure.txt","wb")){
        fwrite($fp,"Prova d'escriptura".PHP_EOL); //final de línia
        fclose($fp);
    } else {
        echo "error obrint el fitxer";
    }
?>
```

## fread

La funció **fread()** llegeix bytes, no cadenes com la funció **fgets()**, de manera que, a diferència de la funció que hem vist anteriorment **fgets()**, que para de llegir quan troba un salt de línia, **fread()** para després d'una quantitat específica de bytes llegits.

```
<?php

    $bytesllegits=fread(descriptorDelArxiu,quantitatBytes);

?>
```

En el següent **exemple** s'aniria llegint cada 10 bytes, incloent salts de línia:

```
<?php
if ($fp=fopen("fitxer.txt", "r")){
    while (!feof($fp)) {
        $linia=fread($fp,10);
        echo $linia."<br/>";
    }
    fclose($fp);
} else {
    echo "Error obrint el fitxer <br/>";
}
?>
```



Veiem la diferència al llegir el mateix arxiu amb `fgets()` i amb `fread()`:

Amb <code>fgets(\$fp)</code>	Amb <code>fread(\$fp,10)</code>
CFGS DAW Entorns de desenvolupament IES Lluís Simarro Xàtiva	CFGS DAW E ntorns de desenvolup ament IES Lluís Sim arro Xàti va

Igual que en el cas de `fwrite()`, per llegir el fitxer s'ha d'obrir incloent el valor `"b"` en el mode d'obertura:

```
$fp=fopen("fitxer.txt", "rb")
```

## file

La funció `file()` permet introduir el contingut d'un fitxer dins d'un array. **Cada fila del fitxer en concret se situarà en un element de l'array consecutiu.** Per exemple:

```
<?php
$linies = file('fitxer.txt'); //$línies serà un array
// Mostrem el fitxer recorrent l'array
foreach ($linies as $num_linia => $linia) {
    echo "Línia ".$num_linia.": " . $linia . "<br/>";
}
?>
```

## file\_get\_contents

La funció `file_get_contents()` retorna el contingut d'un fitxer com una sola cadena (`string`). La **sintaxi bàsica** utilitza un sol paràmetre, encara que en té altres d'opcionals per obtenir parts concretes del fitxer:

```
<?php

    $fitxer=file_get_contents(nomfitxer);

?>
```

```
<?php

    $fitxer = file_get_contents('fitxer.txt'); //fitxer en 1 sol string
    echo $fitxer;

?>
```

## file\_put\_contents

La funció **file\_put\_contents()** escriu dades a un fitxer. Aquesta funció és idèntica a cridar a **fopen()**, **fwrite()** i **fclose()** successivament per a escriure dades en un fitxer.

La **sintaxi bàsica** utilitza dos paràmetres, encara que en té altres d'opcionals anomenats **flags**, per exemple per afegir les dades a un fitxer existent. Si el nom del fitxer no existeix, el fitxer es crea. En cas contrari, se sobreescriu el fitxer existent, a menys que s'establisca el flag FILE\_APPEND:

```
<?php

    file_put_contents(nomfitxer,dades);

?>
```

Per **exemple**:

```
<?php
    $fitxer='fitxer.txt';
    //fiquem tot el fitxer en una cadena
    $contingut_fitxer=file_get_contents($fitxer);
    //afegim un contingut a la cadena que representa el fitxer complet.
    $contingut_fitxer .='Text afegit al final de la cadena';
    //reescrivim el nou contingut al fitxer original
    file_put_contents($fitxer,$contingut_fitxer);
?>
```

Aquest procediment es pot realitzar **amb qualsevol tipus de fitxer** ja siga de text o no.

En php hi ha una gran quantitat de **funcions** relacionades amb el **sistema** de **fitxers**, que permeten manipular fitxers i directoris al servidor. Una llista completa es pot trobar a:

<http://php.net/manual/en/ref.filesystem.php>