



# Lettura e scrittura di file di testo



# HTTP

- HTTP è un protocollo *stateless*, ossia ogni coppia *client-request/server-response* è indipendente dalle altre.
- Tuttavia, per ovviare a questo limite, sono disponibili le seguenti tecniche:
  - Cookie
  - Sessioni
  - File
  - Database



# Lettura e scrittura di file

- fopen(nomeFile, modalità): restituisce un puntatore al file oppure false.
  - 'r': sola lettura
  - 'r+': lettura e scrittura, non elimina il contenuto del file
  - 'w': sola scrittura (se il file non esiste cerca di crearlo)
  - 'w+': lettura e scrittura, elimina il contenuto del file
  - 'a': scrittura in coda
  - ...
- fclose(file): chiude il puntatore a un file aperto



# Lettura e scrittura di file

- fgets(*nomeFile*): restituisce un'intera riga del file oppure *false* in caso di terminazione del file o errore.
- file(*nomeFile*): legge tutto il file e restituisce un array dove ogni elemento è una riga del file; restituisce *false* in caso di errore



# Lettura del file

```
<?php  
    $file = fopen("mioFile.txt", "r");  
  
    if($file)  
    {  
        while(!feof($file))  
        {  
            $line = fgets($file);  
            //gestisco la riga del file  
        }  
  
        fclose($file);  
    }  
?>
```



# Scrittura sul file

```
<?php  
    $file = fopen("mioFile.txt", "w");  
  
    if($file)  
    {  
        fwrite($file, $daScrivereSuFile);  
        fclose($file);  
    }  
?>
```



# Lettura file CSV

- explode(separatore, stringa): restituisce un array spezzando la stringa in corrispondenza del separatore.

```
$fullName = "John Jane Doe";
$pieces = explode(' ', $fullName, 2);
$firstName = $pieces[0]; // John
$lastName = $pieces[1]; // Jane Doe
```

- implode(separatore, array): restituisce una stringa contenente tutti gli elementi separati dal separatore