



- [GUIDE & CONSIGLI](#)
- [PROGETTI](#)

Seleziona una pagina

# I caratteri più utilizzati su .htaccess

# I caratteri più utilizzati su .htaccess

da [Amar A.](#) | Nov 9, 2022 | [PrestaShop](#) | [0 commenti](#)

[Read More](#)

```
01. # Turn on the rewrite engine
02.
03. RewriteEngine On
04. RewriteBase /
05. RewriteRule ^index\.php$ - [L]
06. RewriteCond %{REQUEST_FILENAME} !-f
07. RewriteCond %{REQUEST_FILENAME} !-d
08. RewriteRule . /index.php [L]
09.
10. # Force HTTPS and WWW
11.
12. RewriteCond %{HTTP_HOST} !^www\.($ [OR,NC]
13. RewriteCond %{https} off
14. RewriteRule ^($ https://www/$1 [R=301,L]
15.
16. # Common 301 Redirect Htaccess Rules
17.
18. Redirect 301 /old-link.html /new_link
```

.htaccess



## L'indice dei caratteri utilizzati con .htaccess, una breve guida di riferimento

Questo non è un elenco esaustivo di caratteri, ma piuttosto una sorta di cheat sheet delle espressioni regolari, dei flag e dei codici di stato più comunemente utilizzati.

Per avere una [guida completa ed esaustiva su .htaccess](http://htaccessbook.com) rimandiamo al sito [htaccessbook.com](http://htaccessbook.com).

Character/Flag	Definition
#	Instructs the server to ignore the line. Used for including comments.

<b>Character/Flag</b>	<b>Definition</b>
<b>[F]</b>	Forbidden: instructs the server to return a 403 Forbidden to the client.
<b>[L]</b>	Last rule: instructs the server to stop rewriting after the preceding directive is processed.
<b>[N]</b>	Next: instructs Apache to rerun the rewrite rule until all rewriting is complete.
<b>[G]</b>	Gone: instructs the server to deliver Gone (no longer exists) status message.
<b>[P]</b>	Proxy: instructs server to handle requests by mod_proxy.
<b>[C]</b>	Chain: instructs server to chain the current rule with the previous rule.
<b>[R]</b>	Redirect: instructs Apache to redirect to the specified URL. Note that the default status-code for the [R] flag is 302 (temporary redirect); for permanent redirects use [R=301].
<b>[NC]</b>	No Case: defines any associated argument as case-insensitive.
<b>[PT]</b>	Pass Through: instructs mod_rewrite to pass the rewritten URL for further processing.
<b>[OR]</b>	Or: specifies a logical "or" that ties two expressions together such that either one proving true will cause the associated rule to be applied.
<b>[NE]</b>	No Escape: instructs the server to parse output without escaping characters.
<b>[NS]</b>	No Subrequest: instructs the server to skip the directive if internal sub-request.
<b>[QSA]</b>	Append Query String: directs server to add the query string to the end of the expression.

Character/Flag	Definition
<b>[S=x]</b>	Skip: instructs the server to skip the next "x" number of rules if a match is detected.
<b>[E=var:value]</b>	Environmental Variable: instructs the server to set the variable "var" to "value".
<b>[T=MIME-type]</b>	Mime Type: declares the mime type of the target resource.
<b>[xyz]</b>	Character class: any character within square brackets will be a match. For example, "[xyz]" will match any of the characters x, y, or z.
<b>[xyz]+</b>	Character class in which any combination of items within the brackets will be a match. For example, "[xyz]+" will match any number of x's, y's, z's, or any combination thereof.
<b>[^xyz]</b>	Not within a character class. For example, "[^xyz]" will match any character that isn't x, y, or z.
<b>[a-z]</b>	A dash "-" between two characters within a character class denotes the range of characters between them. For example, "[a-zA-Z]" matches all lowercase and uppercase letters.
<b>a{n}</b>	Exact number, n, of the preceding character, a. For example, "x{3}" matches exactly three x's.
<b>a{n,}</b>	Specifies n or more of the preceding character. For example, "x{3,}" matches three or more x's.
<b>a{n,m}</b>	Specifies a range of numbers, between n and m, of the preceding character, a. For example, "x{3,7}" matches three, four, five, six, or seven x's.

Character/Flag	Definition
( )	Used to group characters together, thereby considering them as a single unit. For example, (htaccess)?book will match "book", with or without the "htaccess" prefix.
^	Denotes the beginning of a regular expression. For example, "^Hello" will match any string that begins with "Hello". Without the caret "^", "Hello" would match anywhere in the string.
\$	Denotes the end of a regular expression. For example, "world\$" will match any string that ends with "world". Without the dollar sign "\$", "world" would match anywhere in the string.
?	Declares as optional the preceding character. For example, "monzas?" will match "monza" or "monzas". In other words, "x?" matches zero or one of "x".
!	Declares negation. For example, "!string" matches everything except "string".
.	A literal dot (or period) indicates any single arbitrary character.
-	Instructs Apache to NOT rewrite the URL. Example syntax: "example.com - [F]"
+	Matches one or more of the preceding character. For example, "G+" matches one or more G's, while "+" will match one or more characters of any kind.
*	Matches zero or more of the preceding character. For example, use ".*" as a wildcard.
	Declares a logical "or" operator. For example, "(x y)" matches "x" or "y".

<b>Character/Flag</b>	<b>Definition</b>
<code>\</code>	Escape special characters such as: <code>^ \$ ! . *   ( ) [ ] { }</code>
<code>\.</code>	Indicates a literal dot (escaped).
<code>/*</code>	Zero or more slashes.
<code>.*</code>	Zero or more arbitrary characters.
<code>^\$</code>	Defines an empty string.
<code>^.*\$</code>	The standard pattern for matching everything.
<code>[^/.]</code>	Defines one character that is neither a slash nor a dot.
<code>[^/.]+</code>	Defines any number of characters that contains neither slash nor dot.
<code>http://</code>	This is a literal statement – in this case, the literal character string, “http://”.
<code>^example.*</code>	Matches a string that begins with the term “example”, followed by any character(s).
<code>^example\.com\$</code>	Defines the exact string, “example.com”.
<code>-d</code>	Tests if string is an existing directory.
<code>-f</code>	Tests if string is an existing file.
<code>-s</code>	Tests if file in test string has a non-zero value.

[← Previous Post](#) [Next Post →](#)



Written by Amar A.

## CONTATTO

Nome

Telefono

Mail

Messaggio

Invia

**Articoli nella medesima categoria**





## **eCommerce e vendita online: guida introduttiva**

Feb 7, 2023

Guida introduttiva 2023 all'eCommerce e vendita online, consigli utili per la tua attività su web e sulla scelta della piattaforma di vendita.

[leggi tutto](#)



## **eCommerce e vendita online: guida introduttiva**

Feb 7, 2023

Guida introduttiva 2023 all'eCommerce e vendita online, consigli utili per la tua attività su web e sulla scelta della piattaforma di vendita.

[leggi tutto](#)



## eCommerce e vendita online: guida introduttiva

Feb 7, 2023

Guida introduttiva 2023 all'eCommerce e vendita online, consigli utili per la tua attività su web e sulla scelta della piattaforma di vendita.

[leggi tutto](#)

### 0 Comments

0 commenti

### Invia un commento

Il tuo indirizzo email non sarà pubblicato. I campi obbligatori sono contrassegnati \*

Commento \*

Nome \*

Email \*

Sito web

Invia commento

□

## Email

hello@easyseopro.it

↵

## Orari

Qualsiasi, siamo su web potete contattarci quando desiderate, risponderemo prima possibile.

□

## Privacy Policy

Vedi la nostra [privacy policy](#) conforme GDPR.