

Apache Configuration File

*# (!) Using `.htaccess` files slows down Apache, therefore, if you have access
to the main server config file (usually called `httpd.conf`), you should add
this logic there: <http://httpd.apache.org/docs/current/howto/htaccess.html>.*

#Allow from 127.0.0.1/21

*# #####
CROSS-ORIGIN RESOURCE SHARING (CORS)
#####*

*# -----
| Cross-domain AJAX requests
-----*

*# Enable cross-origin AJAX requests.
<http://code.google.com/p/html5security/wiki/CrossOriginRequestSecurity>
<http://enable-cors.org/>*

*# <IfModule mod_headers.c>
Header set Access-Control-Allow-Origin "*"
</IfModule>*

*# -----
| Web fonts access
-----*

Allow access from all domains for web fonts

*# <IfModule mod_headers.c>
<FilesMatch "\.(eot|font\.css|otf|ttc|ttf|woff)\$">
Header set Access-Control-Allow-Origin "*"
</FilesMatch>
</IfModule>*

*# #####
ERRORS
#####*

```
# -----
# | 404 error prevention for non-existing redirected folders
# -----

# Prevent Apache from returning a 404 error for a rewrite if a directory
# with the same name does not exist.
# http://httpd.apache.org/docs/current/content-negotiation.html#multiviews
# http://www.webmasterworld.com/apache/3808792.htm
```

Options -MultiViews

```
# #####
# AUTHENTICATION AND SECURITY
# #####

# -----
# | File access
# -----

# Block access to directories without a default document.
# Usually you should leave this uncommented because you shouldn't allow anyone
# to surf through every directory on your server (which may includes rather
# private places like the CMS's directories).

<IfModule mod_autoindex.c>
    Options -Indexes
</IfModule>

# -----

# Block access to hidden files and directories.
# This includes directories used by version control systems such as Git and SVN.

<IfModule mod_rewrite.c>
    RewriteCond %{SCRIPT_FILENAME} -d [OR]
    RewriteCond %{SCRIPT_FILENAME} -f
    RewriteRule "(^/)\." - [F]
</IfModule>

# -----

# Block access to hidden files and directories.
# This includes directories used by version control systems such as Git and SVN.
```

```
# Block access to backup and source files.
# These files may be left by some text editors and can pose a great security
# danger when anyone has access to them.

<FilesMatch "(^#.#|\.|bak|config|dist|fla|htaccess|htpasswd|incl|ini|log|old|phps|psd|shl|sql
  Order Allow,Deny
  Deny from all
  Satisfy All
</FilesMatch>

# -----
# | Limit upload file size to protect against DOS attack ### #bytes, 0-2147483647(2GB)
# -----

LimitRequestBody 10240000

# -----
# | Server software information
# -----

# Avoid displaying the exact Apache version number, the description of the
# generic OS-type and the information about Apache's compiled-in modules.

# ADD THIS DIRECTIVE IN THE `httpd.conf` AS IT WILL NOT WORK IN THE `.htaccess`!

# ServerTokens Prod

# #####
# DEFAULTS
# #####

Options -ExecCGI
DirectoryIndex index.php index.html index.htm
ServerSignature Off

# -----

DefaultLanguage en-US

# -----
# | UTF-8 encoding
```

```
# -----  
  
# Use UTF-8 encoding for anything served as `text/html` or `text/plain`.  
AddDefaultCharset utf-8  
# Or AddType 'text/html; charset=UTF-8' html  
  
# Force UTF-8 for certain file formats.  
<IfModule mod_mime.c>  
    AddCharset utf-8 .atom .css .js .json .rss .vtt .webapp .xml  
</IfModule>  
  
# -----  
# I Proper MIME types for all files  
# -----  
  
<IfModule mod_mime.c>  
  
    # Audio  
    AddType audio/mp4 m4a f4a f4b  
    AddType audio/ogg oga ogg  
  
    # JavaScript  
    # Normalize to standard type (it's sniffed in IE anyways):  
    # http://tools.ietf.org/html/rfc4329#section-7.2  
    AddType application/javascript js jsonp  
    AddType application/json json  
  
    # Video  
    AddType video/mp4 mp4 m4v f4v f4p  
    AddType video/ogg ogv  
    AddType video/webm webm  
    AddType video/x-flv flv  
    AddType video/x-f4v f4v  
    # Switch off gzipping for video files, can prevent playback or streaming in some browsers  
  
    # Web fonts  
    AddType application/font-woff woff  
    AddType application/vnd.ms-fontobject eot  
  
    # Browsers usually ignore the font MIME types and sniff the content,  
    # however, Chrome shows a warning if other MIME types are used for the
```

following fonts.

AddType application/x-font-ttf ttc ttf

AddType font/opentype otf

Make SVGZ fonts work on iPad:

<https://twitter.com/FontSquirrel/status/14855840545>

AddType image/svg+xml svg svgz

AddEncoding gzip svgz

Other

AddType application/octet-stream safariextz

AddType application/x-chrome-extension crx

AddType application/x-opera-extension oex

AddType application/x-shockwave-flash swf

AddType application/x-web-app-manifest+json webapp

AddType application/x-xpinstall xpi

AddType application/xml atom rdf rss xml

AddType image/webp webp

AddType image/x-icon ico

AddType text/cache-manifest appcache manifest

AddType text/vtt vtt

AddType text/x-component htc

AddType text/x-vcard vcf

</IfModule>

*# -----
| Force download - FilesMatch "\.(?i:csv|xls|doc|pdf|avi|mov|mp4|mv4|mp3|mpe?g|jpe?g|png)|
-----*

Force file to download instead of appear in browser

Via mime type

AddType application/octet-stream .avi .mov .mp3 .mp4 .mv4 .mpg .mpeg .pdf .csv .xls .doc .

Via file

<IfModule mod_headers.c>

<FilesMatch "\.(?i:csv|xls|doc|pdf)\$">

ForceType application/octet-stream

Header set Content-Disposition "attachment"

Allow from all

</FilesMatch>

```
# </IfModule>

# -----
# | Better website experience for Internet Explorer
# -----

# Force IE to render pages in the highest available mode in the various
# cases when it may not: http://hsivonen.iki.fi/doctype/ie-mode.pdf.
# Use, if installed, Google Chrome Frame.

<IfModule mod_headers.c>
    Header set X-UA-Compatible "IE=edge,chrome=1"
    # `mod_headers` can't match based on the content-type, however, we only
    # want to send this header for HTML pages and not for the other resources
    <FilesMatch "\.(appcache|crx|css|eot|gif|htc|ico|jpe?
gl|js|m4a|m4v|manifest|mp4|oex|ogal|oggl|ogv|otf|pdf|png|safariextz|svg|svgz|ttf|vcl|webapp|wet
    Header unset X-UA-Compatible
    </FilesMatch>

    <FilesMatch "\.(html|html|php)$">
        Header set imagetoolbar "no"
    </FilesMatch>

</IfModule>

# #####
# URL REWRITES
# #####

# -----
# | Rewrite engine
# -----

# Turning on the rewrite engine and enabling the `FollowSymLinks` option is
# necessary for the following directives to work.

# If your web host doesn't allow the `FollowSymLinks` option, you may need to
# comment it out and use `Options +SymLinksIfOwnerMatch` but, be aware of the
# performance impact: http://httpd.apache.org/docs/current/misc/perf-tuning.html#symlinks

# Also, some cloud hosting services require `RewriteBase` to be set:
```

http://www.rackspace.com/knowledge_center/frequently-asked-question/why-is-mod-rewrite-not-working

```
<IfModule mod_rewrite.c>
  Options +FollowSymLinks
  # Options -MultiViews
  # Options +SymLinksIfOwnerMatch
  RewriteOptions MaxRedirects=15

  # Start rewriting
  RewriteEngine On
  # Use a base if the site URL is based on a specific site root directory,
  #RewriteBase /_work-kit/codeigniter-kit/

  # Disable automatic directory detection
  DirectorySlash Off

  # Generate rewrite base dynamically
  RewriteCond %{REQUEST_URI}::$1 ^(\.*/)(.*)::\2$
  RewriteRule ^(\.*/)$ - [E=BASE:%1]
  #%{ENV:BASE}

  # Set application environment; helps to ignore rules like https in development
  # http://stackoverflow.com/questions/10231324/apache-mod-rewrite-force-www-only-if-not-in-development
  #RewriteCond %{HTTP_HOST} ^localhost$
  #RewriteRule .* - [E=APPLICATION_ENVIRONMENT:development,NE]
  #RewriteCond %{HTTP_HOST} ^127\.0\.0\.1$
  #RewriteRule .* - [E=APPLICATION_ENVIRONMENT:development,NE]
  #RewriteCond %{REMOTE_ADDR} ^127\.0\.0\.1$
  #RewriteRule .* - [E=APPLICATION_ENVIRONMENT:development,NE]
  #RewriteCond %{REMOTE_ADDR} ^::1$
  #RewriteRule .* - [E=APPLICATION_ENVIRONMENT:development,NE]
  #%{ENV:APPLICATION_ENVIRONMENT}
  #RewriteCond %{ENV:APPLICATION_ENVIRONMENT} !=development [NC]
  # Set application environment; helps to ignore rules like https in development
  # http://stackoverflow.com/questions/19362746/rewritecond-with-setenv
  SetEnvIfNoCase HOST 42cdev.com APPLICATION_ENVIRONMENT=development
  SetEnvIfNoCase HOST localhost APPLICATION_ENVIRONMENT=development
  SetEnvIfNoCase HOST 127.0.0.1 APPLICATION_ENVIRONMENT=development
  SetEnvIfNoCase REMOTE_ADDR 127.0.0.1 APPLICATION_ENVIRONMENT=development
  SetEnvIfNoCase REMOTE_ADDR ::1 APPLICATION_ENVIRONMENT=development
  #%{ENV:APPLICATION_ENVIRONMENT}
```

```
#RewriteCond %{ENV:APPLICATION_ENVIRONMENT} !=development [NC]

# Try to fix mysql admin issues
#RewriteCond %{HTTP_HOST} ^(mysql)\.+$ [NC]
#RewriteRule (.*) $1 [L]

# -----

# Rewrite rules to block out some common exploits
## If you experience problems on your site block out the operations listed below
## This attempts to block the most common type of exploit 'attempts' to Joomla!
#
# Block out any script trying to base64_encode crap to send via URL
RewriteCond %{QUERY_STRING} base64_encode.*\(.*\) [OR]
# Block out any script that includes a <script> tag in URL
RewriteCond %{QUERY_STRING} (\<| %3C).*script.*(\>| %3E) [NC,OR]
# Block out any script trying to set a PHP GLOBALS variable via URL
RewriteCond %{QUERY_STRING} GLOBALS(=| \[| \\\[0-9A-Z]{0,2}) [OR]
# Block out any script trying to modify a _REQUEST variable via URL
RewriteCond %{QUERY_STRING} _REQUEST(=| \[| \\\[0-9A-Z]{0,2})
# Send all blocked request to homepage with 403 Forbidden error!
RewriteRule ^(.*)$ %{ENV:BASE}index.php [F,L]

# -----

# Prevent Codeigniter directory access with 403 Forbidden error
RewriteCond %{REQUEST_URI} ^(ci|_)/.*
RewriteRule ^(.*)$ %{ENV:BASE}index.php/$1 [F,L]
#RewriteRule ^(.*)$ %{ENV:BASE}index.php?/$1 [F,L]

# -----

# -----
# | Suppressing / Forcing the "www." at the beginning of URLs
# -----

# The same content should never be available under two different URLs especially
# not with and without "www." at the beginning. This can cause SEO problems
# (duplicate content), therefore, you should choose one of the alternatives and
# redirect the other one.
```



```
# By default option 1 (no "www.") is activated:
# http://no-www.org/faq.php?q=class_b

# If you'd prefer to use option 2, just comment out all the lines from option 1
# and uncomment the ones from option 2.

# IMPORTANT: NEVER USE BOTH RULES AT THE SAME TIME!

# Option 1: rewrite www.example.com to example.com
#RewriteCond %{HTTPS} !=on
#RewriteCond %{HTTP_HOST} ^www\.(.+\.)$ [NC]
#RewriteRule ^ http://%1%{REQUEST_URI} [R=301,L]

# Option 2: rewrite example.com to www.example.com
# Be aware that the following might not be a good idea if you use "real"
# subdomains for certain parts of your website.
# If you have subdomains, you can add them to the list using the "|" (OR) regex operator
#RewriteCond %{HTTPS} !=on
#RewriteCond %{HTTP_HOST} !^(www|subdomain)\.(.+\.)$ [NC]
#RewriteRule ^ http://www.%{HTTP_HOST}%{REQUEST_URI} [R=301,L]

# Redirect certain pages
#Redirect 301 /buying-a-mattress/support /best-mattress-for-you/support
# or
#RewriteCond %{REQUEST_URI} buying-a-mattress/support$
#RewriteRule ^(.*)$ %{ENV:BASE}best-mattress-for-you/support [R=301,L]

# Enforce www on https
#RewriteCond %{HTTPS} on
#RewriteCond %{HTTP_HOST} !^(www) [NC]
#RewriteRule ^(.*)$ https://www.%{HTTP_HOST}%{REQUEST_URI} [L,R=301]

# Ignore if in development
#RewriteCond %{ENV:APPLICATION_ENVIRONMENT} !=development [NC]
# Enforce https
#RewriteCond %{HTTPS} !=on
#RewriteRule ^(.*)$ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]

# Ignore if in development
#RewriteCond %{ENV:APPLICATION_ENVIRONMENT} !=development [NC]
# Enforce https on CloudFlare
```

```

# Enable support on CloudFlare
#RewriteCond %{HTTP: CF-Visitor} '"scheme": "http"'
#RewriteRule ^(.*)$ https://www.domain.com/$1 [L]
#RewriteRule ^(.*)$ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]

# Ignore if in development
#RewriteCond %{ENV: APPLICATION_ENVIRONMENT} !=development [NC]
# Enforce www
#RewriteCond %{HTTP_HOST} !^(www|subdomain)\.+$ [NC]
#RewriteRule ^ https://www.%{HTTP_HOST}%{REQUEST_URI} [L,R=301]

# -----

# -----

# | Filename-based cache busting
# -----

# If you're not using a build process to manage your filename version revving,
# you might want to consider enabling the following directives to route all
# requests such as `/css/style.12345.css` to `/css/style.css`.

# To understand why this is important and a better idea than `*.css?v231`, read:
# http://stevesouders.com/blog/2008/08/23/revving-filenames-dont-use-querystring

# <IfModule mod_rewrite.c>
#   RewriteCond %{REQUEST_FILENAME} !-f
#   RewriteCond %{REQUEST_FILENAME} !-d
#   RewriteRule ^(.+)\.(\d+)\. (js|css|png|jpg|gif)$ $1. $3 [L]
# </IfModule>

#RewriteRule ^__ver[0-9]+__/(.*)$ /$1 [QSA]

RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^(.+)\.(\d+)\. (js|css|png|jpg|gif)$ %{ENV:BASE}$1. $3 [NC,L]

# -----

# Normal Rewrites
#RewriteCond %{REQUEST_URI} !/$
#RewriteRule ^industry-news/(.*)$ industry-news\.php/$1 [NC,L]
#RewriteRule ^(.*)$ https://www.domain.com/$1 [L,R=301]

```

```

#RewriteRule wellness/\.(.*)$ wellness\.php/$1 [NC,L]

# Rewrite PPC Landing pages
#RewriteCond %{REQUEST_FILENAME} !-f
#RewriteRule ^(.*)-store-lander$ ppc/landing\.php?store=$1-something&ab=c [NC]
#RewriteRule ^(.*)-ppc-lander-?(.*)$ ppc/landing\.php?store=$1&ab=$2 [NC,L]

# -----

# Redirect to existing file
RewriteCond %{REQUEST_URI} !/$
RewriteCond %{REQUEST_FILENAME}.php -f
RewriteRule (.*) %{ENV:BASE}$1\.php [L]
#RewriteRule ^.*(testing).*$ http://google.com?regex=$1&r_f=%{REQUEST_FILENAME}&d_r=%{DOCUMENT_ROOT}&s_f=%{SCRIPT_FILENAME}&b=%{ENV:BASE} [L]

# Removes trailing slashes (prevents SEO duplicate content issues); RewriteBase was neces:
#RewriteCond %{REQUEST_FILENAME} !-d
#RewriteRule ^(.+)/$ /$1 [L,R=301]
##
#RewriteCond %{REQUEST_FILENAME} !-d
#RewriteCond %{REQUEST_URI} (.+)/$
#RewriteRule ^ %1 [L,R=301]
##
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^(.+)/$ %{ENV:BASE}$1 [L,R=301]

# Enforce trailing slash
#RewriteCond %{REQUEST_FILENAME} !-f
#RewriteRule ^(.+[^/])$ /$1/ [L,R=301]

# Redirect to external file
#RewriteCond %{REQUEST_FILENAME} !-f
#RewriteRule ^(images/.*)$ http://www.example.com/client/site.com/$1 [L,R=301,NC]

# -----

# Force download
RewriteCond %{QUERY_STRING} download=1
RewriteRule ([^/]*\.(gif|jpg|jpeg|png|mp3|mp4))$ - [L,E=DOWNLOAD_FILE:$1]
Header onsuccess set Content-disposition "attachment; filename=%{DOWNLOAD_FILE}.e" env=DOWN

```

```
# -----

## If the file or directory DOES NOT exist
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d

## Permanent redirects
#RewriteRule ^emerge(.*)$ %{ENV:BASE}connect/young-adults$1 [R=301,L,QSA]
#Redirect 301 /connect/kids(.*) http://www.example.com/connect/children/$1
#Redirect 301 /staff.php http://www.example.com/about/team/

## Error
#RewriteRule ^(.*)$ %{ENV:BASE}404.php?url=$1 [L,QSA]

## Route to the front controller
RewriteRule ^(.*)$ %{ENV:BASE}index.php/$1 [L,QSA]
#RewriteRule ^(.*)$ index.php/$1 [L,QSA]
#RewriteRule ^(.*)$ index.php?/$1 [L,QSA]

</IfModule>

# -----
# | Secure Sockets Layer (SSL)
# -----

# Rewrite secure requests properly to prevent SSL certificate warnings, e.g.:
# prevent `https://www.example.com` when your certificate only allows
# `https://secure.example.com`.

# <IfModule mod_rewrite.c>
#   RewriteCond %{SERVER_PORT} !^443
#   RewriteRule ^ https://example-domain-please-change-me.com%{REQUEST_URI} [R=301,L]
# </IfModule>

# -----

# Force client-side SSL redirection.

# If a user types "example.com" in his browser, the above rule will redirect him
# to the secure version of the site. That still leaves a window of opportunity
```

```
# (the initial HTTP connection) for an attacker to downgrade or redirect the
# request. The following header ensures that browser will ONLY connect to your
# server via HTTPS, regardless of what the users type in the address bar.
# http://www.html5rocks.com/en/tutorials/security/transport-layer-security/

# <IfModule mod_headers.c>
#   Header set Strict-Transport-Security max-age=16070400;
# </IfModule>

# -----
# | Custom error messages / pages
# -----

# You can customize what Apache returns to the client in case of an error (see
# http://httpd.apache.org/docs/current/mod/core.html#error-document), e.g.:

#ErrorDocument 404 /404.html

# -----
# | Errors after rewriting
# -----

<IfModule !mod_rewrite.c>
    # Without mod_rewrite, route 404's to the front controller
    #ErrorDocument 404 /404.html
</IfModule>

# #####
# WEB PERFORMANCE
# #####

# -----
# | Content transformations
# -----

# Prevent some of the mobile network providers from modifying the content of
# your site: http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.9.5.

# <IfModule mod_headers.c>
#   Header set Cache-Control "no-transform"
# </IfModule>
```

```

# -----
# | GZip Compression
# -----

<IfModule mod_deflate.c>

    # Force compression for mangled headers.
    # http://developer.yahoo.com/blogs/ymn/posts/2010/12/pushing-beyond-gzipping
    <IfModule mod_setenvif.c>
        <IfModule mod_headers.c>
            SetEnvIfNoCase ^(\Accept-Encoding X-cept-Encoding X(15)|^(15)|-(15))$ ^((gzip|deflate)\
HAVE_Accept-Encoding
            RequestHeader append Accept-Encoding "gzip,deflate" env=HAVE_Accept-Encoding
        </IfModule>
    </IfModule>

    # Compress all output labeled with one of the following MIME-types
    # (for Apache versions below 2.3.7, you don't need to enable `mod_filter`
    # and can remove the `<IfModule mod_filter.c>` and `</IfModule>` lines
    # as `AddOutputFilterByType` is still in the core directives).
    #<IfModule mod_filter.c>
    AddOutputFilterByType DEFLATE \
    application/atom+xml \
    application/javascript \
    application/json \
    application/rss+xml \
    application/vnd.ms-fontobject \
    application/x-font-ttf \
    application/x-javascript \
    application/x-web-app-manifest+json \
    application/xhtml+xml \
    application/xml \
    font/opentype \
    image/svg+xml \
    image/x-icon \
    text/css \
    text/html \
    text/javascript \
    text/plain \
    text/x-component \

```

```
text/xml
```

```
# Netscape 4.x has some problems...
```

```
BrowserMatch ^Mozilla/4 gzip-only-text/html
```

```
# Netscape 4.06-4.08 have some more problems
```

```
BrowserMatch ^Mozilla/4\.0[678] no-gzip
```

```
# MSIE masquerades as Netscape, but it is fine
```

```
BrowserMatch \bMSIE !no-gzip !gzip-only-text/html
```

```
# NOTE: Due to a bug in mod_setenvif up to Apache 2.0.48
```

```
# the above regex won't work. You can use the following
```

```
# workaround to get the desired effect:
```

```
BrowserMatch \bMSIE[!] !no-gzip !gzip-only-text/html
```

```
# Don't compress images, already zipped files, or pdfs
```

```
SetEnvIfNoCase Request_URI \
```

```
  \.(?:gif|jpe?g|png)$ no-gzip dont-vary
```

```
SetEnvIfNoCase Request_URI \
```

```
  \.(?:exe|t?gz|zip|bz2|sit|rar)$ no-gzip dont-vary
```

```
SetEnvIfNoCase Request_URI \.pdf$ no-gzip dont-vary
```

```
# Switch off gzipping for video files, can prevent playback or streaming in some browsers
```

```
SetEnvIfNoCase Request_URI \.(og[gv]|mp4|m4v|webm)$ no-gzip dont-vary
```

```
#</IfModule>
```

```
</IfModule>
```

```
# Make sure proxies don't deliver the wrong content
```

```
# This caused an error for me, so I'm commenting it out.
```

```
<IfModule mod_headers.c>
```

```
  Header append Vary User-Agent env=! dont-vary
```

```
</IfModule>
```

```
# -----
```

```
# I ETag removal
```

```
# -----
```

```
# Since we're sending far-future expires headers (see below), ETags can
```

```
# be removed: http://developer.yahoo.com/performance/rules.html#etags.
```

```
# `FileETag None` is not enough for every server.
```

```
<IfModule mod_headers.c>
```

```
  Header unset ETag
```

```
</IfModule>
```

```
FileETag None
```

```
# -----  
# | Expires headers (for better cache control)  
# -----
```

```
# The following expires headers are set pretty far in the future. If you don't  
# control versioning with filename-based cache busting, consider lowering the  
# cache time for resources like CSS and JS to something like 1 week.
```

```
<IfModule mod_expires.c>
```

```
ExpiresActive on
```

```
ExpiresDefault "access plus 1 month"
```

```
# CSS
```

```
ExpiresByType text/css "access plus 1 year"
```

```
# Data interchange
```

```
ExpiresByType application/json "access plus 0 seconds"
```

```
ExpiresByType application/xml "access plus 0 seconds"
```

```
ExpiresByType text/xml "access plus 0 seconds"
```

```
# Favicon (cannot be renamed!)
```

```
ExpiresByType image/x-icon "access plus 1 week"
```

```
# HTML components (HTCs)
```

```
ExpiresByType text/x-component "access plus 1 month"
```

```
# HTML
```

```
ExpiresByType text/html "access plus 0 seconds"
```

```
# JavaScript
```

```
ExpiresByType application/javascript "access plus 1 year"
```

```
# Manifest files
```

```
ExpiresByType application/x-web-app-manifest+json "access plus 0 seconds"
```

```
ExpiresByType text/cache-manifest "access plus 0 seconds"
```

```
# Media
```



```
ExpiresByType audio/ogg "access plus 1 week"
ExpiresByType image/gif "access plus 1 week"
ExpiresByType image/jpeg "access plus 1 week"
ExpiresByType image/png "access plus 1 week"
ExpiresByType video/mp4 "access plus 1 week"
ExpiresByType video/ogg "access plus 1 week"
ExpiresByType video/webm "access plus 1 week"
```

Web feeds

```
ExpiresByType application/atom+xml "access plus 1 hour"
ExpiresByType application/rss+xml "access plus 1 hour"
```

Web fonts

```
ExpiresByType application/font-woff "access plus 1 month"
ExpiresByType application/vnd.ms-fontobject "access plus 1 month"
ExpiresByType application/x-font-ttf "access plus 1 month"
ExpiresByType font/opentype "access plus 1 month"
ExpiresByType image/svg+xml "access plus 1 month"
```

```
<IfModule mod_headers.c>
```

Set up caching on media files for 1 year

```
<FilesMatch "\.(ico|flv|pdf|mov|mp3|wmv|ppt)$">
    ExpiresDefault "access plus 1 year"
    Header append Cache-Control "public"
</FilesMatch>
```

Set up caching on media files for 1 week

```
<FilesMatch "\.(gif|jpg|jpeg|png|swf)$">
    ExpiresDefault "access plus 1 week"
    Header append Cache-Control "public, proxy-revalidate"
</FilesMatch>
```

Set up 1 Hour caching on commonly updated files

```
<FilesMatch "\.(xml|txt|html|html|js|css)$">
    ExpiresDefault "access plus 1 hour"
    Header append Cache-Control "private, proxy-revalidate, must-revalidate"
</FilesMatch>
```

Force no caching for dynamic files

```
<FilesMatch "\.(pl|php|cgi|ssl|scgi|fcgi)$">
```

```
ExpiresDefault "access plus 0 seconds"
Header set Cache-Control "no-cache, no-store, must-revalidate, max-age=0, proxy-revali
Header set Pragma "no-cache"
#ExpiresDefault "access plus 1 hour"
#Header append Cache-Control "private, proxy-revalidate, must-revalidate"
</FilesMatch>

</IfModule>

</IfModule>

# -----
# | Filename-based cache busting
# -----

# If you're not using a build process to manage your filename version revving,
# you might want to consider enabling the following directives to route all
# requests such as `/css/style.12345.css` to `/css/style.css`.

# To understand why this is important and a better idea than `*.css?v231`, read:
# http://stevesouders.com/blog/2008/08/23/revving-filenames-dont-use-querystring

# <IfModule mod_rewrite.c>
#   RewriteCond %{REQUEST_FILENAME} !-f
#   RewriteCond %{REQUEST_FILENAME} !-d
#   RewriteRule ^(.+)\.(\d+)\.(\.js|.css|.png|.jpg|.gif)$ $1.$3 [L]
# </IfModule>

# #####
# PHP SETTINGS
# #####

# php_value upload_max_filesize 20M
# php_value post_max_size 20M
# php_value max_execution_time 300
# php_value max_input_time 600
# php_value memory_limit 96M
```