Files represent a significant part of the data that browsers load when rendering web pages. Common examples of files are images, and web resources such as CSS stylesheets, JavaScript code files or page attachments. The system provides two types of file caching mechanisms:

- **Client-side**
- **Server-side**

File caching works for the CSS stylesheet objects used in Kentico, even if they are not stored as physical files.

We also recommend using using code minification and compression.

Client-side file caching

Client-side caching duplicates the data of previously requested files directly within browser applications or other clients (such as intermediate network caches). Client cache is the most efficient type of caching, because it allows browsers to access files without communicating with the web server. The system controls client caching by modifying the HTTP headers of file request responses.

To configure client-side file caching:

1. Open the **Settings** application.
2. Select the **System -> Performance** category.
3. Edit the settings in the **Client caching** category:

Setting	Description
Client cache (minutes)	Sets the number of minutes for which clients consider cached files to be valid and up-to-date (i.e. the length of the client cache expiration time). Specifically, the setting determines how the server sets the **Expires** response header for file requests. The following values are possible: • **0** - cached files are always flagged as expired, so clients revalidate files on each request. Does NOT disable client caching for files. • **More than 0** - until a cached file expires, clients load the file's data from the cache without sending any requests to the server. Provides the best file performance, but may cause browsers to display outdated content (if cached files are modified before the expiration interval ends). This setting only applies to dynamic files that are managed by the Kentico application, for example stylesheets in the *CSS stylesheets* application. Physical files stored on the file system may use a different client cache expiration time: • Physical files served by the Kentico application through the *GetResource.ashx* handler use the **CMSPhysicalFilesCacheMinutes** web.config key. The default value is one week. • Files that are requested directly use the IIS client cache configuration.
Allow client cache revalidation	If enabled, the server adds the **must-revalidate** directive to the **Cache-control** response header of file requests. Ensures that caching clients always perform revalidation when requesting expired files (most clients revalidate by default even without the *must-revalidate* directive). We only recommend disabling revalidation if you don't need the site visitors to receive updated version of the content they have loaded before.

4. **Save** the settings.

Clients (browsers) store files in their cache after requesting them for the first time (with an expiration time specified by the settings). Once the file expires, the client sends a *revalidation* request to the server, which can have the following results:

- If the server contains a newer version of the file than the client cache, the revalidation fails and the server returns the full file (standard 200 HTTP response).

- If the file is unchanged, the server returns a 304 Not Modified response. The client cache keeps the file and refreshes the expiration time. The client and server only exchange HTTP header information, which is significantly faster than fully reloading files.

Server-side file caching

When a client requests an image or other file, Kentico stores the file's data in the application memory.

- For files stored in the **database**, the cache holds the entire file, including binary data.
- For files stored in the **file system**, the cache stores only the main file records. Binary data is streamed directly from the file system.

The system can then serve subsequent requests for the file without needing to access the database or file system.

To set how long the server-side cache retains files:

1. Open the **Settings** application.
2. Select the **System -> Performance** category.
3. Type a number of minutes into the **Cache files (minutes)** setting. We recommend a value between 1 and 60 minutes (depending on the size of the files used on your website and the server's available memory).
4. **Save** the settings.

Files stay in the cache for the specified number of minutes.

⚠ **Note**

- File caching cannot cause the website to display outdated content in most scenarios. The cache automatically removes files when they are modified.

- If your project uses external file storage (such as Azure Blob Storage or Amazon S3), you need to clear the cache manually after making changes to files or wait for the cache to expire. The automatic clearing of the server-side file cache does not work for files in external storage.

ℹ If your website stores large files in the database, you can limit caching of files that exceed a certain size. Specify the threshold by typing a number of kilobytes into the **Maximum file size to cache** setting. Files larger than the specified value are only cached for 1 minute, regardless of the *Cache files (minutes)* setting.