

An important factor that influences the performance of a website is the size of code resources, which are requested by client browsers when rendering pages. The most significant among these are CSS stylesheets and JavaScript source files, which can in some cases reach sizes that considerably impact page load time.

Kentico provides two types of builtin functionality that help optimize the performance of requests for CSS and JavaScript resources:

Code minification	<p>Removes all unnecessary characters from the code that are not required by the browser to correctly process the resource. This includes white spaces, line break characters, comments, bookmarks, etc.</p> <p>Minified resources behave in exactly the same way as the original. Browsers can immediately handle minified resources without any additional steps.</p> <p>Note:</p> <ul style="list-style-type: none"> • Minified code is difficult to read (for humans) and is therefore unsuitable for debugging. • JavaScript minification does not work on instances running on Microsoft Azure Cloud Services (even if the corresponding setting is enabled).
Resource compression	<p>Reduces the size of resources by encoding their data. To apply compression, you must also enable minification for resources of the given type.</p> <p>When a client browser receives a compressed resource, it must first decompress the data. All browsers officially supported by Kentico should be able to correctly decompress files. In cases where the client cannot process compressed data, the system automatically sends the unmodified resource instead.</p>

You can enable or disable code minification and compression globally for all sites in **Settings -> System -> Performance**, through the settings in the **Resources** category.

Reducing the size of requested resources saves bandwidth and improves the response time of your website. Minification can decrease the size of a resource by approximately 20–40%, depending on the code of the given object. If compression is also used, resources can be reduced to roughly 30% of their original size.



Note: The reductions only apply to resources stored individually in the file system or database. Inline code inserted into the HTML markup of pages is not affected.

The minification and compression process slightly increases the server CPU load and adds a short delay to resource requests. To counter this issue, the system uses server-side caching for resource files:

- Minification/compression only occurs once when a resource is requested for the first time (the cache stores the result).
- On subsequent client requests, the system provides the transformed version of the resource from the cache.
- The cache stores both compressed and uncompressed versions of resources, so the data is readily available even for clients without compression support.
- You can set the number of minutes for which files are cached through the **Cache files (minutes)** setting in **Settings -> System -> Performance**.

Additionally, you can use clientside browser caching for resources (enabled by default). With client caching, browsers only reload resources from the server if the cached data expires or the content of the resource becomes outdated. See [Caching files and resources](#) to learn how to configure the client-side file caching.



Using external preprocessing tools

You can also use external preprocessing tools (LESS, Grunt tasks, ...) and store your CSS and JS files as static physical files. You lose the ability to edit the files within Kentico UI, but avoid having the system to compile the resources with the first request each time the cache expires.

Requesting minified resources

Compression and minification is automatically ensured by the `~/CMSPages/GetResource.ashx` handler, which manages resource requests according to the specified settings. If minification is enabled, CSS and JavaScript requests generated by the system use this handler.

You can manually load resources in your code by using the following URL parameters with the handler:

- **stylesheetname** - used to request a [CSS stylesheet](#) from the database. The code name of the requested stylesheet must be entered as the value of the parameter.
Link example:
`<link href="/Kentico/CMSPages/GetResource.ashx?stylesheetname=CorporateSite" type="text/css" rel="stylesheet">`
- **_transformations, _layouts, _templates, _devicelayouts, _webparts, _webpartlayouts, _containers** - used to request the internal stylesheets of the corresponding type of page component. The object ID values of the given components must be entered as the value of the parameter, separated by semicolons (if multiple component stylesheets are requested). See [Adding CSS to page components](#).
Link examples:
`<link href="/Kentico/CMSPages/GetResource.ashx?_containers=1;14" type="text/css" rel="stylesheet"/>`
`<link href="/Kentico/CMSPages/GetResource.ashx?_transformations=3511" type="text/css" rel="stylesheet">`
- **stylesheetfile** - used to request static CSS resources from the file system. The relative path of the requested .css file must be entered into the parameter.
Link example:
`<link href="/Kentico/CMSPages/GetResource.ashx?stylesheetfile=/Kentico/App_Themes/Design/OnSiteEdit.css" type="text/css" rel="stylesheet">`
- **scriptfile** - used to request static JavaScript resources from the file system. The relative path of the requested .js file must be entered into the parameter.
Link example:
`<script src="/Kentico/CMSPages/GetResource.ashx?scriptfile=%7e%2fCMSScripts%2fmootools.js" type="text/javascript"></script>`


Resource requests with minification disabled

If minification is disabled, the system generates requests with a direct URL (for JavaScript files) or using the `~/CMSPages/GetCSS.aspx` system page (for CSS stylesheets).

Requests in this URL format are always supported, but they do not perform minification or compression of resources.

Enabling compression of page output

You can configure the system to compress the HTML output code of all pages rendered by Kentico.

 **Note:** Output compression requires compliance from client browsers. Browsers that do not support compression always download uncompressed page output.

1. Go to **Settings -> System -> Performance**.
2. Check **Enable output compression**.
3. Click **Save**.

The setting applies globally for all sites in the system.