

Preview mode in Kentico provides a way to view the latest version of pages before they are published (for example when using [Workflow](#)). If you wish to use preview mode in an MVC application that handles the presentation of the website's pages, you need to implement the required functionality in the application's code.

Before you can use preview URLs, you need to [specify the URL pattern](#) (in Kentico administration) for your content only page types. The URL pattern is then used to create URLs to the Kentico content presented by an external application. Preview URL for pages are generated:

- When using the main **Preview** mode in the **Pages** application.
- When generating preview links for pages in **Pages -> Properties -> General -> Preview URL**. See: [Sending links to unpublished pages](#)

The preview URLs the system generates for content only pages consist of additional information, such as language version and a hash for validating the URL. The Kentico [integration package](#) in your MVC application validates and processes the preview URL. The package then removes the additional information to leave only the original presentation URL based on the URL pattern. The original presentation URL then goes through the standard routing process.



**Note:** You need to manually ensure that your MVC application displays the **latest versions of pages** in preview mode. See [Loading the latest versions of pages](#) for details.

## Enabling preview mode in MVC applications

Preview mode support is provided in the [Kentico.Content.Web.Mvc integration package](#). To enable preview support in your MVC application:

1. If necessary, set up your Kentico instance and MVC application (see [Starting with MVC development](#)).
2. [Specify a URL pattern](#) for content only pages.
3. [Install](#) the **Kentico.Content.Web.Mvc** integration package in your MVC project.
4. Enable the preview feature in the MVC application's **ApplicationConfig.cs** file:

```
using Kentico.Content.Web.Mvc;

...

public static void RegisterFeatures(ApplicationBuilder builder)
{
    ...
    builder.UsePreview();
    ...
}
```



### Framing policy and preview mode

By default, the MVC application sends the [X-Frame-Options HTTP header](#) with the *SAMEORIGIN* framing policy. The [Kentico integration package](#) disables the header for preview URLs. This allows users to preview content handled by the MVC application in the Kentico administration interface, even when running the administration on a different domain.

## Excluding tracking scripts for preview mode

If you use third-party analytics or tracking scripts on your website (for example Google Tag Manager scripts), we strongly recommend that you disable them for preview mode. The preview URLs are intended for internal use within the Kentico administration interface or for viewing unpublished pages. Exposing preview URLs to third parties and having them affect your website's analytics data is typically not desired.

To exclude tracking scripts for preview mode, use conditions that [evaluate whether preview mode is enabled for requests](#). You can add the conditions directly around the script code within your site's views, or within related business code that you use to include the scripts.

## Checking whether preview mode is enabled for requests

Use the following code to check whether preview mode is enabled for a request:

```
using Kentico.Content.Web.Mvc;

...

bool previewEnabled = System.Web.HttpContext.Current.Kentico().Preview().Enabled;
```

The location where you need to check the request context for preview mode depends on the implementation of your MVC application. For example, directly in the code of your controllers or when registering repositories that make use of your [generate d providers](#).



**Tip:** If the *URL pattern* that you use for your content only pages does not include a variable for the page culture (for example if you use a different mechanism to pass the requested culture), you can get the culture from the context of the preview request:

```
using Kentico.Content.Web.Mvc;

...

string cultureName = System.Web.HttpContext.Current.Kentico().Preview().
CultureName;
```

## Loading the latest versions of pages

For requests that have preview mode enabled, you need to ensure that your code loads the **latest** versions of pages instead of the *published* versions. For more information about retrieving page data, see [Generating classes for Kentico objects](#). For example:

```
// Uses DocumentQuery to load article pages, getting the latest versions instead of
the published versions
var articles = ArticleProvider.GetArticles()

                                                                    .LatestVersion(true)
                                                                    .Published(false)
                                                                    .OnSite("mvcsite")
                                                                    .Culture("en-us")
                                                                    .OrderByDescending

( "DocumentPublishFrom" );
```

If you use [data caching](#) for your pages, you need to disable it for preview mode. This ensures that caching works correctly for standard requests, but is not used for the different content displayed in preview mode.



### Preview mode and controller output caching

If you use caching of controller output in your MVC application ([OutputCache](#) attributes), the [Kentico integration package](#) in your MVC application automatically ensures that the output cache is not used for page requests in preview mode.



Preview mode should now work correctly for the Kentico application that you use to manage your website's content.



See our [MVC Demo site](#) for an advanced example of an MVC application that uses preview mode.