Kentico allows you to write your own SQL search provider, which you can use to integrate a third-party search engine. You can also create a custom search provider if you need to modify or filter search results returned by the standard search engine.

You cannot customize the search provider in the App_Code folder. The modifications must be added as part of a new assembly, which you then need to register via the **CMSSearchProviderAssembly** web.config key.

This provider only affects the SQL Search engine. The Smart Search engine remains unchanged.

To modify the smart search, create and use custom search indexes.

Adding the custom search provider assembly

The following example demonstrates how to customize the SQL search provider so that it only searches a limited section of the website.

1. Open your Kentico solution in Visual Studio.
2. Add a new **Class Library** project to the Kentico solution, for example named *CustomSearchProvider*.
3. Add references to the required Kentico libraries (DLLs) for the new project:
    a. Right-click the project and select **Add -> Reference**.
    b. Select the **Browse** tab of the **Reference manager** dialog, click **Browse** and navigate to the ***Lib*** folder of your Kentico web project.
    c. Add references to the following libraries:
        - **CMS.Base.dll**
        - **CMS.Core.dll**
        - **CMS.DataEngine.dll**
        - **CMS.Helpers.dll**
        - **CMS.Search.dll**
        - **CMS.SearchProviderSQL.dll**
4. Reference the *CustomSearchProvider* project from the Kentico web project *(CMSApp* or *CMS)*.
5. Open your Kentico program files directory (by default *C:\Program Files\Kentico\<version>*) and expand the *CodeSamples\CustomizationSamples\CustomSearchProvider* subfolder.
6. Copy the **SearchProvider.cs** file into the *CustomSearchProvider* directory in your web project**.**
7. Include the new file into the *CustomSearchProvider* project in Visual Studio:
    a. Expand the *CustomSearchProvider* project in the Solution Explorer.
    b. Click **Show all files** at the top of the Solution Explorer.
    c. Right-click the *SearchProvider.cs* file and select **Include in Project**.
8. Edit the *SearchProvider* class and rename the namespace to **exactly** match the assembly name of your custom project (*CustomSearchProvider* in this example).
9. Build the **CustomSearchProvider** project (Build the entire solution on web application projects).

The CustomSearchProvider project is now integrated into your application.

Modifying the search provider class

1. Edit the **SearchProvider.cs** class in the **CustomSearchProvider** project.
2. Modify the code of the **Search** method according to the following:

```
public DataSet Search(string siteName, string searchNodePath, string cultureCode,
string searchExpression, SearchModeEnum searchMode, bool searchChildNodes, string
classNames, bool filterResultsByReadPermission, bool searchOnlyPublished, string
whereCondition, string orderBy, bool combineWithDefaultCulture)
{
        // Limits the search path to only the /Products section of the website
        searchNodePath = "/Products/%";

        // Gets a new instance of the SQL SearchProvider
        CMS.SearchProviderSQL.SearchProvider limitedSearchProvider = new CMS.
SearchProviderSQL.SearchProvider();

        // Gets the search result dataset based on the parameters of the Search
method and the limited search path

        DataSet searchResults = limitedSearchProvider.Search(siteName,
searchNodePath, cultureCode, searchExpression, searchMode, searchChildNodes,
classNames, filterResultsByReadPermission, searchOnlyPublished, whereCondition,
orderBy, combineWithDefaultCulture);

        return searchResults;
}
```

3. Save the file and **Rebuild** the CustomSearchProvider project.

In real customization scenarios, you can modify the **Search** method to behave according to any requirements. The method must always return a DataSet containing the search results.

Registering the custom search provider

1. Edit your application's web.config file.
2. Add the following key to the configuration/appSettings section:

```
<add key="CMSSearchProviderAssembly" value="CustomSearchProvider" />
```

The system loads the **SearchProvider** class from the assembly specified as the value of the **CMSSearchProviderAssembly** key.

Result

To try out the customized search engine, open the live website and search for text using the SQL Search engine. For example, you can use the **Examples -> Web Parts -> FullText Search -> SQL Search -> SQL Search dialog with results** page on the sample Corporate site.

The search now only returns results from the /Products section of the website.

This customization example is purely demonstrative. If you need to change the search path, directly set the Path property of individual SQL search web parts or controls.