Customization of the data provider allows you to implement your own database connector.

You cannot modify the data provider in the App_Code folder. The customizations must be added as part of a new assembly, which you then need to register via the **CMSDataProviderAssembly** web.config key.

> Custom data providers are **not** intended for accessing non-Microsoft SQL Server database engines. They only allow you to modify how queries are executed against the Microsoft SQL Server.

Adding the custom data provider assembly

1. Open your Kentico solution in Visual Studio.
2. Add a new **Class Library** project to the Kentico solution, for example named *CustomDataProvider*.
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**
4. Reference the *CustomDataProvider* 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\CustomDataProvider* subfolder.
6. Copy the following files into the *CustomDataProvider* directory in your web project:
    - **DataConnection.cs**
    - **DataProvider.cs**
    - **SqlGenerator.cs**
    - **TableManager.cs**
7. Include the new files into the *CustomDataProvider* project in Visual Studio:
    a. Expand the *CustomDataProvider* project in the Solution Explorer.
    b. Click **Show all files** at the top of the Solution Explorer.
    c. Select the new files while holding the **Ctrl** key.
    d. Right-click one of the files and select **Include in Project**.
8. Edit the copied files and rename the namespaces to **exactly** match the assembly name of your custom project (*CustomDataProvider* in this example).
9. Build the **CustomDataProvider** project (Build the entire solution on web application projects).

The CustomDataProvider project is now integrated into your application. By default, the classes in the custom data provider are identical to the ones used by default, but you can modify them according to your own requirements.

Registering the custom data provider

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

```
<add key="CMSDataProviderAssembly" value="CustomDataProvider"/>
```

The system loads the **DataProvider** class from the assembly specified as the value of the **CMSDataProviderAssembly** key.

Your application now uses the custom data provider for handling of database operations.