**How to Troubleshoot FakeDb Errors - Sitecore 8.2**

A NuGet package, [Sitecore.FakeDb](https://github.com/sshushliapin/Sitecore.FakeDb) helps us to test functions that use [Sitecore](https://www.sitecore.com/) API. It is a mock memory database where you can remove the dependencies and isolate the code. When you create a brand-new test project, setting up FakeDb can be tricky.

Installing FakeDb and adding the usual two Sitecore dll, Sitecore.Kernel.dll and Sitecore.Mvc.dll will not work. The best resources for getting FakeDb up and running for Sitecore 8.2 are:

* [FakeDb Installation](https://github.com/sshushliapin/Sitecore.FakeDb/wiki/Installation)
* [Sitecore 8.2 – No constructor found error](https://github.com/sshushliapin/Sitecore.FakeDb/issues/139)

These documentations are great. But, what really helps us is the checklist that we can go through whenever we creates test projects that use FakeDb.

**FakeDb configuration checklist**

When you set up a test project with FakeDb, ask yourself these questions. It will save time for troubleshooting.

1. **Did you add references to the assemblies below?**

<pre>

Lucene.Net.dll

Sitecore.Analytics.dll

Sitecore.Kernel.dll

Sitecore.Logging.dll

Sitecore.Nexus.dll

</pre>

1. **Did you set the correct path to your license file in App.config (in the root of the test project)?**

<pre>

<sitecore>

<settings>

<setting name="LicenseFile" value="..\..\..\libs\license.xml" />

</settings>

…

</sitecore>

</pre>

<script src="https://gist.github.com/mydatahack/3140f83229252d2fa8ec0349006a3e9f.js"></script>

1. **For Sitecore 8.2 (or maybe 9 and above), did you add Sitecore.FakeDb.config in the App\_Config/Include folder?**

<script src="https://gist.github.com/mydatahack/a440f3f5ceefde856edc3ac5a40a0eb6.js"></script>

Ok, If you have done all of the steps above, FakeDb should work. Just in case you forgot, these are the possible errors.

**Potential Errors**

1. **Object reference set to null**

When you try to instantiate FakeDb without the correct dependencies, you will get object reference set to null error with no helpful nested error message. Don’t freak out. Just add the references.

1. **License required error**

FakeDb requires Sitecore license to run. This error may not be obvious. If your path to the license file is wrong, you get this error when the function calls Sitecore API like GetItem().

If you wrap the block where you are calling Sitecore API with a try-catch block, this error may not appear when you run unit test. Instead, your unit test will fail from the actual different from the expected. When you put a debugger, you will see it is failing at Sitecore API call and is being caught by the catch clause. Tricky, but easy to fix. Just correct the license path.

1. **No matching constructor was found**

You may get the error as below:

<pre>

Sitecore.Exceptions.ConfigurationException

: Could not create instance of type

: Sitecore.Data.Database. No matching constructor was found.

</pre>

To fix this, you need to add the below config file in App\_Config/Include/Sitecore.FakeDb.config as described [here](https://github.com/sshushliapin/Sitecore.FakeDb/issues/139).

The better option is to add this config file in Foundation.Testing project and refer it from all the test projects. This is exactly what [Habitat](https://github.com/Sitecore/Habitat) does. You can actually see it has the FakeDb config file in the foundation project.

I would like to hear from you if you still have trouble running FakeDb. I can probably help you!