Tutorial 1: Setting up DirectX 11 with Visual Studio

Before writing any graphics code we'll need to have the tools to do so. The first of these tools is a compiler that is preferably built into a nice IDE. The one I use and will be supplying project files for is Visual Studio 2015. There are several others available and some are free off the net. I'll leave that up to you to decide which one you prefer. I personally use Visual Studio 2015 Community because it is free and is an excellent IDE. You can download Visual Studio 2015 Community from the Visual Studio website. When you install Visual Studio 2015 make sure to choose custom and select all so that the Visual C++ components are all installed, otherwise it sets it up primarily for C# development.

The second tool you will need is the Windows 10 SDK. The Windows 10 SDK contains all of the DirectX 11 headers, libraries, DLLs, and so forth that you will need to write DirectX 11 applications. If you installed Visual Studio 2015 then the SDK was already installed with it. Otherwise you can download the Windows Standalone SDK for Windows 10 from Windows Dev Center (msdn) website. After downloading and installing the SDK you will have the files necessary to compile DirectX 11 programs. The documentation for the Windows 10 SDK is all on the Windows Dev Center website as well. You will find the Direct3D 11 Programming Guide there which has all the DirectX 11 documentation as well as some example code.

Once you have the IDE and the SDK installed you can now setup your IDE to work with the Windows 10 SDK so that you can write DirectX 11 applications. Please note that some IDEs will need to be installed first before installing the Windows 10 SDK.

Setting Up Visual Studio 2015

In Visual Studio 2015 I used the following steps:

First you need to create an empty Win32 project so select File -> New -> Project. In the New Project menu select Win32 under Templates -> Visual C++ -> Win32. Then select Win32 Project from the choices. Give the project a name (I called mine Engine) and a location and then click on "Ok". Click "Next" and you will be given another menu. Under "Additional options" put a check mark in the "Empty project" box and unclick the Security Development Lifecycle (SDL) checks. Click on "Finish" and you should now have an basic Win32 empty project setup.

Then on the top bar you will see the value "x86" in the Solution Platforms dropdown, select this and pick "x64" instead. This will set your project to 64bit instead of the default 32bit.

Summary

Our DirectX 11 development environment should now be setup and ready for us to start writing DirectX 11 applications.

To Do Exercises

1. Have a quick look over the DirectX 11 Programming Guide on the Windows Dev Center (msdn) website.

Back to Tutorial Index