# Setting up Code::Blocks with SDL.

This tutorial was created by Paul Wortmann.
You may freely distribute this file under the CC-BY-SA 3.0  license.
E-mail : physhex@gmail.com
Website: www.physhexgames.co.nr


The goal of this tutorial is to setup the Code::Blocks IDE with SDL.
We will also be installing MinGW. The programming language covered in this tutorial is C/C++.
And we will be compiling 32bit code. Whether you are using a 64bit or 32bit Windows version is
irrelevant  with regards to this tutorial.
For a lot of people new to programming, setting up their first IDE and API to work together can be
challenging. It's not that it is a difficult task to accomplish, it's due to the fact that it is easy to do
once you know how, and only needs to be done once, and as such there is sometimes a lack of
information available to assist with these issues.
What we really want is to just start programming and not be pulling our hair out over a trivial
matter of installing and setting up our development environment.


Ok, so this tutorial will be broken up into 5 sections:
1.  Installing the Code::Blocks IDE.
2.  Installing the latest Code::Blocks "nightly".
3.  Installing Mingw.
4.  Configuring Code::Blocks to use MinGW.
5.  Installing SDL, SDL_mixer, SDL_image and SDL_ttf.

# 1. Installing the Code::Blocks IDE.

Firstly we need to start by downloading the Code::Blocks IDE.
At the time of writing the latest version of Code::Blocks is Version 10.05.
While this version is stable and incorporates  many features, it is 2 years old....
The Code::Blocks development team is tirelessly  working on the next stable version, and often
release "nightly" builds. (Usually once a month) The current nightly build has far more features and
capabilities than that of the last stable release. So after we install version 10.05, I will show you
how to install the latest nightly build, this however is an optional step.  The nightly builds are
development snapshots and may not be as stable as version 10.05. Although I have been using
nightly builds for a few years now, and I find them to be very stable worth the install.
You can download Code::Blocks version 10.05 from the Code::Blocks website
http://www.codeblocks.org/downloads/26
For the purpose of this tutorial we will be installing the one called codeblocks-10.05-setup.exe



After running the downloaded executable, we will be presented with a series of windows to guide
us through the installation  process. Basically we are just clicking next and doing a full installation
into the default installation directory. Someone new to computers or who would like to be 100%
sure they are installing it correctly may find the following series of images to be helpful. If however
you are familiar with Code::Blocks or have already installed it, then proceed to the next section.
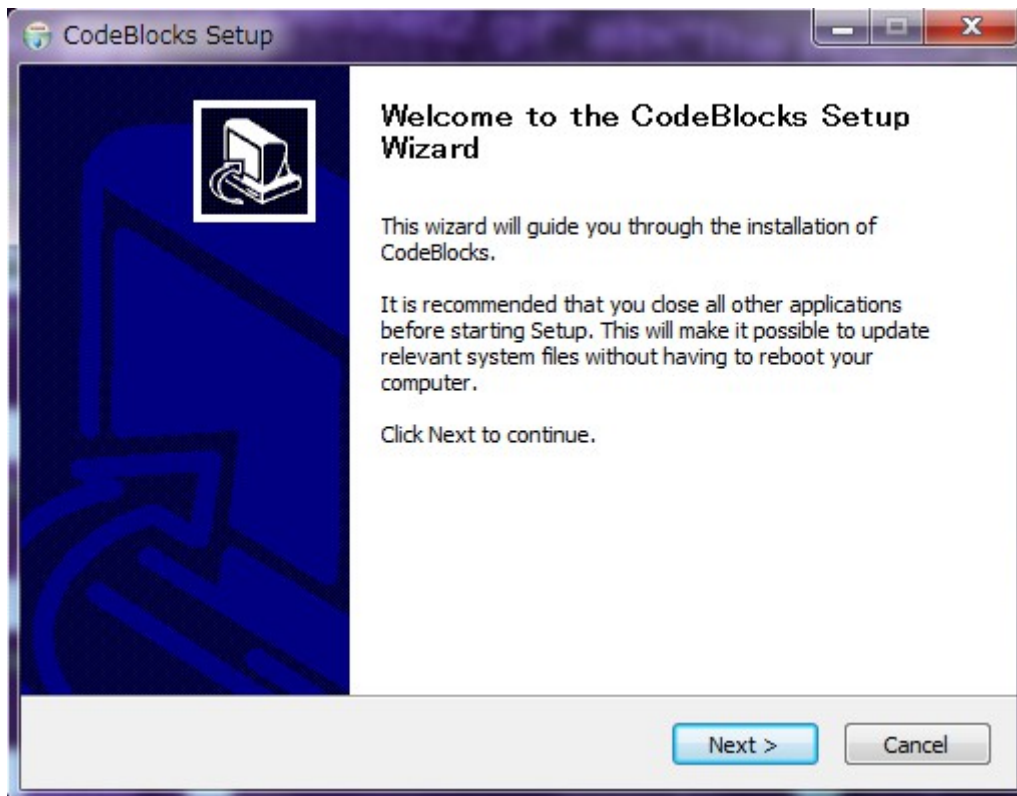
Step 1: After running the executable we are presented with this window.
Although I never closed any of my open applications when I clicked next, it may be best to head the advice, and then click "next".
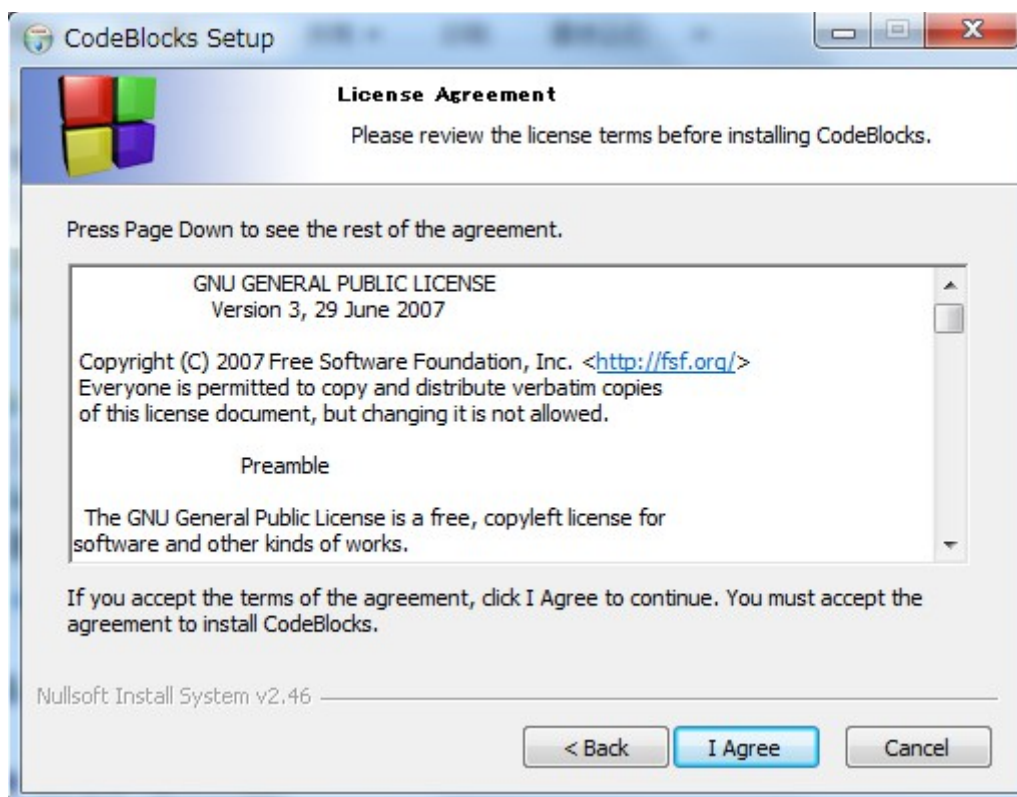


Step 2: After carefully reading the license agreement, either click "I agree" or "Cancel".
If you do choose to click on the "Cancel" button, I would advise you to also stop reading and delete this tutorial, as it will no longer be beneficial to you. If you do click on "I agree" then you  may proceed with this tutorial.
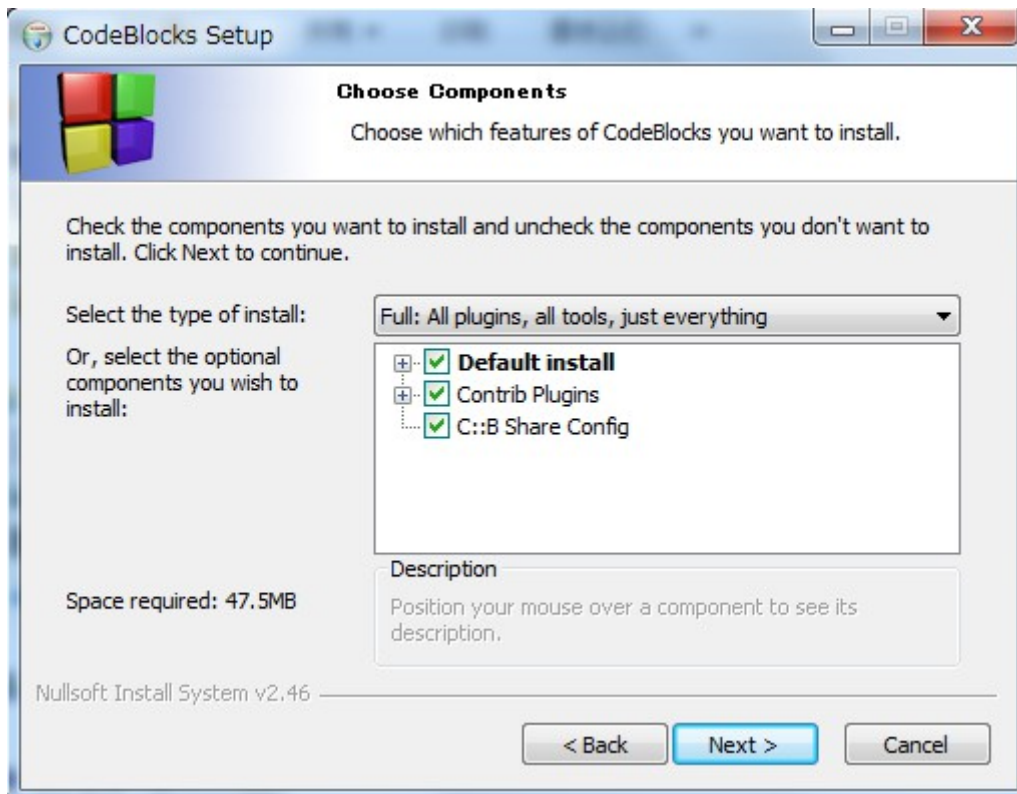
Step 3: Although we may not require all the features at this moment in time, we may however find them useful in the future, so we will be selecting a Full install from the drop-down menu and then clicking "next".
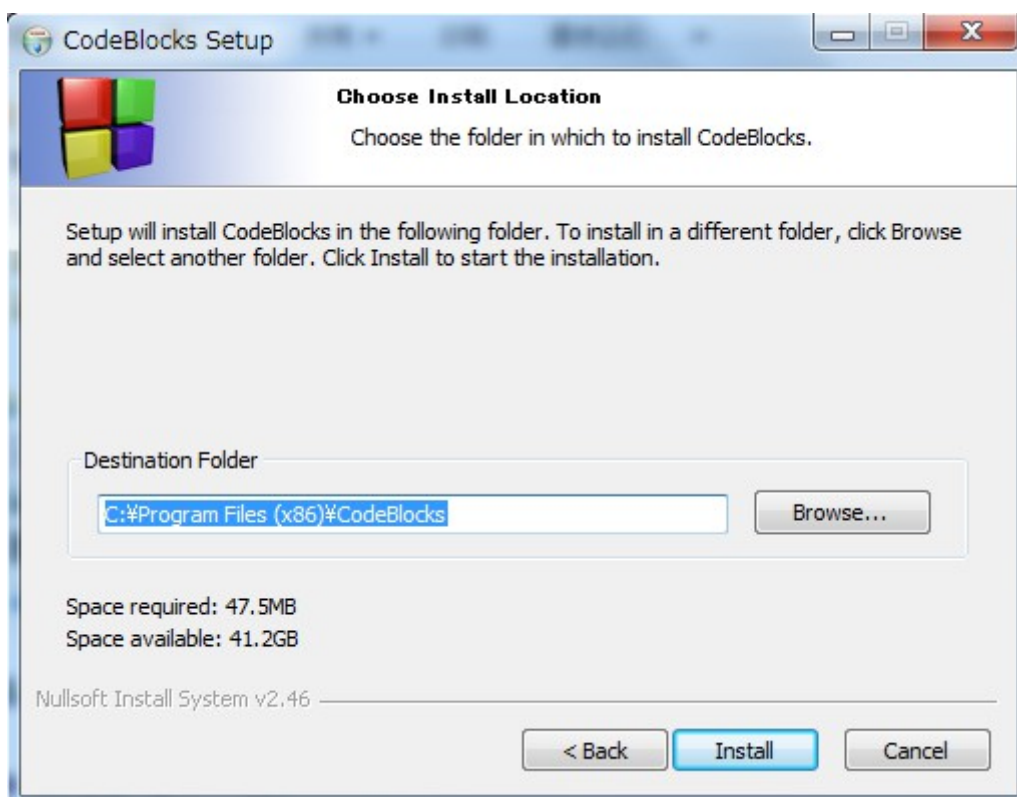


Step 4: On the next window we will be asked to specify the destination folder for our Code::Blocks installation. You may change the default path, but for the purpose of this tutorial we will be installing Code::Blocks into the default installation folder. So just click "Install".
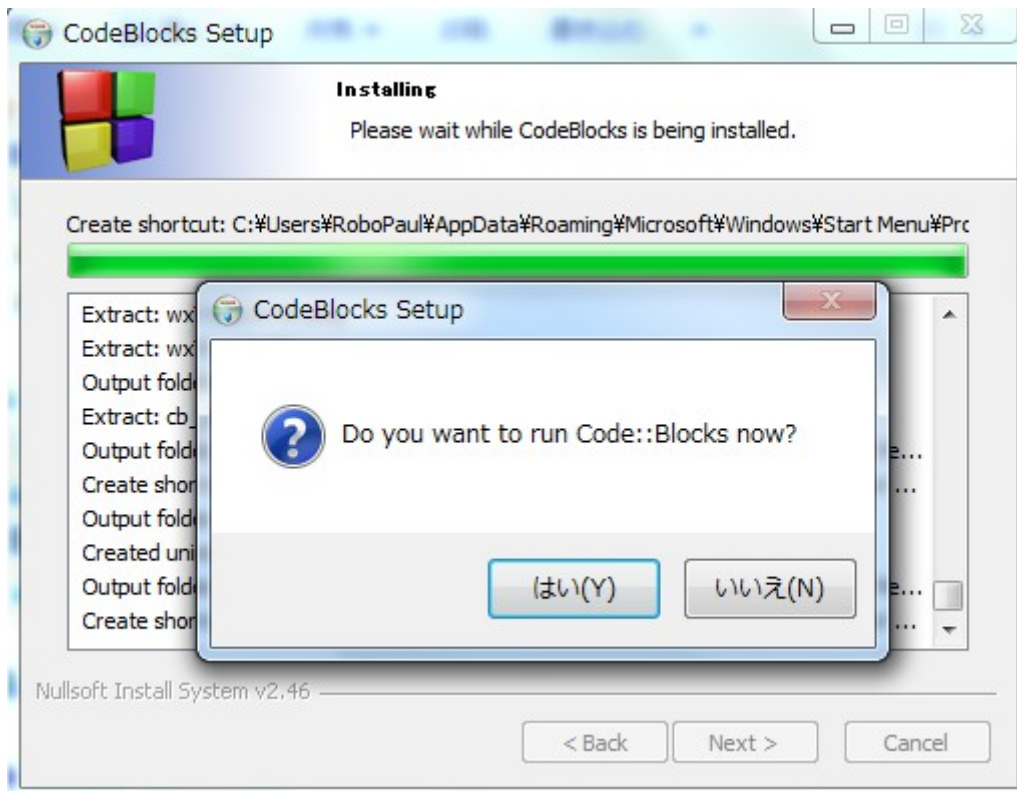
Step 5: While we could probably run Code::Blocks now, it wont do us any use until we have installed the other components we require. So we will just be clicking "(N)" or "No".
I live in Japan and therefore have a Japanese version of windows installed, so yours should
be       in English, or the default language of your operating system.
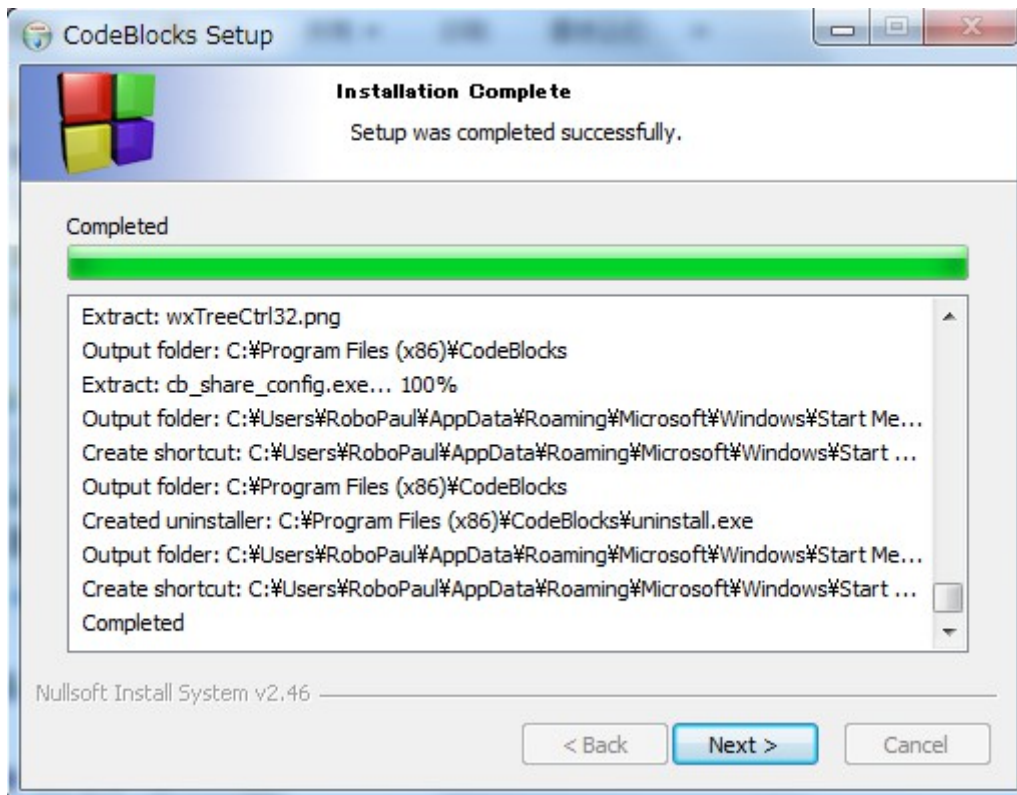


Step 6: The installation is almost complete, so just just click "Next".

Step 7: The installation is now compete and you can click "Finish".

## 2. Installing the latest Code::Blocks "nightly".

While this step is optional, I do recommend it. Having the latest build will equip you with a bountiful selection of new and improved features.
Code::Blocks "nightly" builds can be located on the Code::Blocks forum.
http://forums.codeblocks.org/index.php/board,20.0.html
At the time of writing the latest build is 04 April 2012 build (7918)
You will need to download all 3 files:
The unicode windows wxWidget dll for Code::Blocks
The mingw10m.dll
The 04 April 2012 build (7918) Code::Blocks binary.

Once you have downloaded these 3 files, extract them and replace the existing files with the new ones. Ie. Overwrite the files in your "C:\Program Files (x86)\CodeBlocks\" directory with the new files.
*How to extract files and use your operating system is beyond the scope of this tutorial. If you are having difficulty, remember Google is your friend.

# 3. Installing Mingw.

What we have done so far is install only the IDE itself, so we need to install a compiler and configure Code::Blocks to use it.
The compiler we will be using is the mingw port of the GNU compiler suite.
You can find more information on their site here http://www.mingw.org/
Now it is not completely obvious at first what we actually need to download and install, so we will be using the automated installer "mingw-get". It can be easily located on the mingw sourceforge page: http://sourceforge.net/projects/mingw/
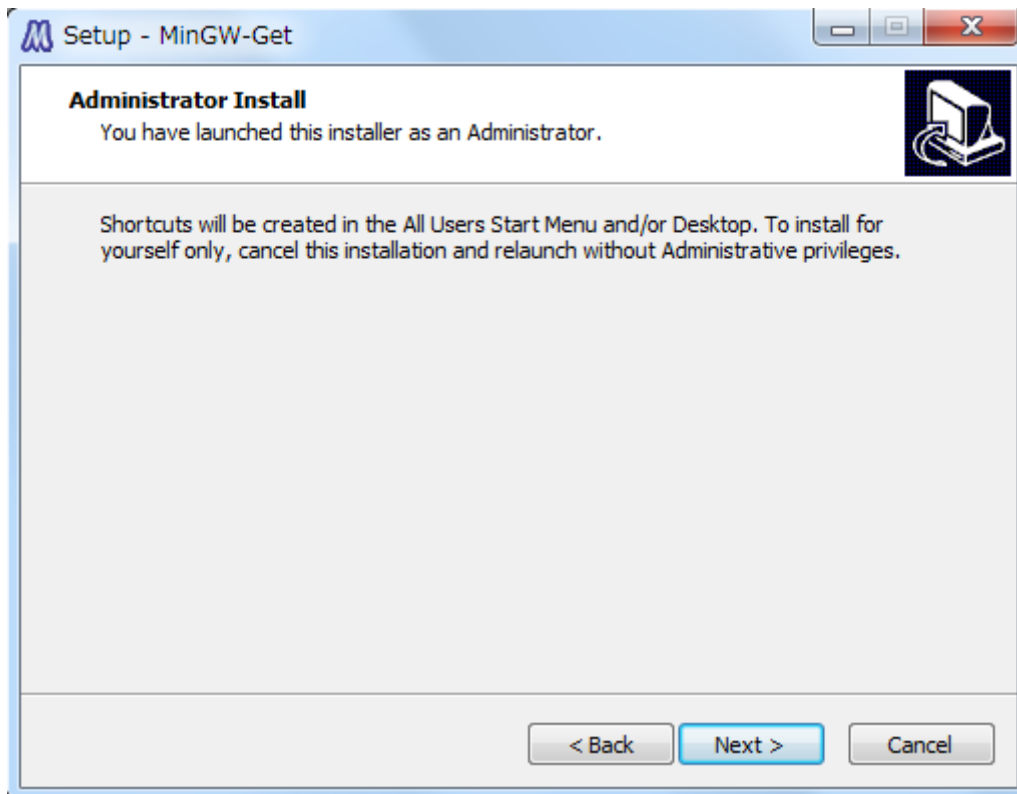At the time of writing the latest build was mingw-get-inst-20111118.exe
After downloading this file proceed with the following steps:
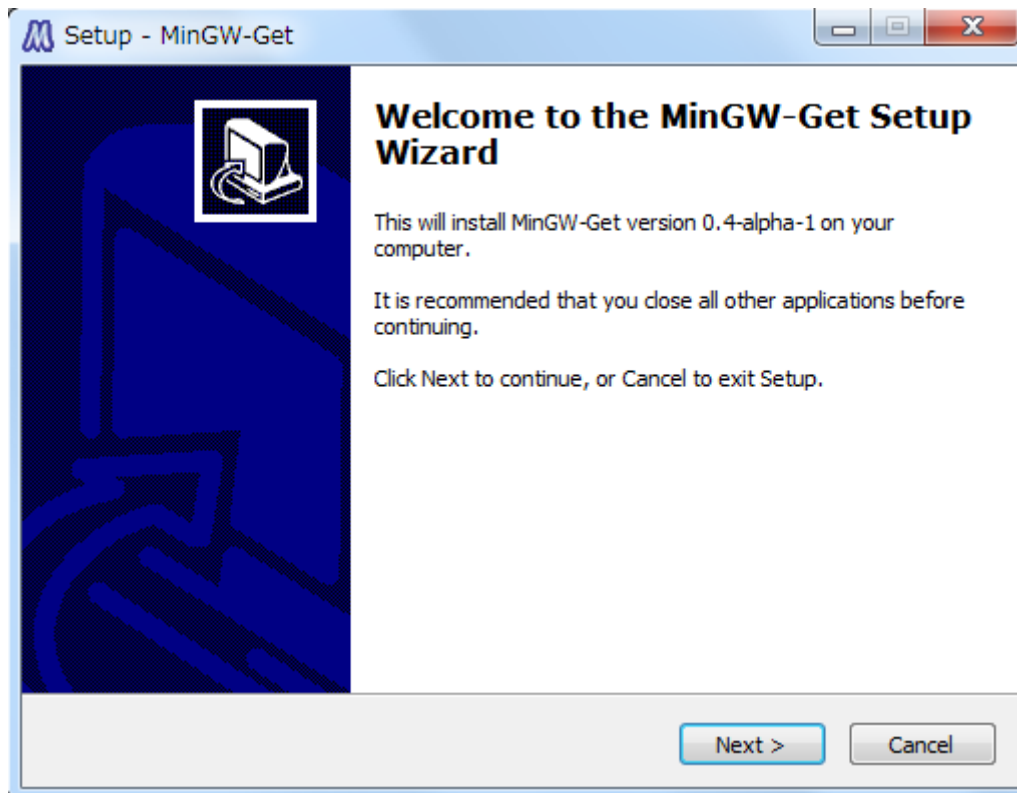


Step 1: After running the application file, we will be presented with the following window. You may proceed to click "next", after you have closed other applications that may be running, although to be honest I didn't close anything.

Step 2: As I am running this installation as an administrator, this window was shown, which has advice for single user installation. I just clicked "next".
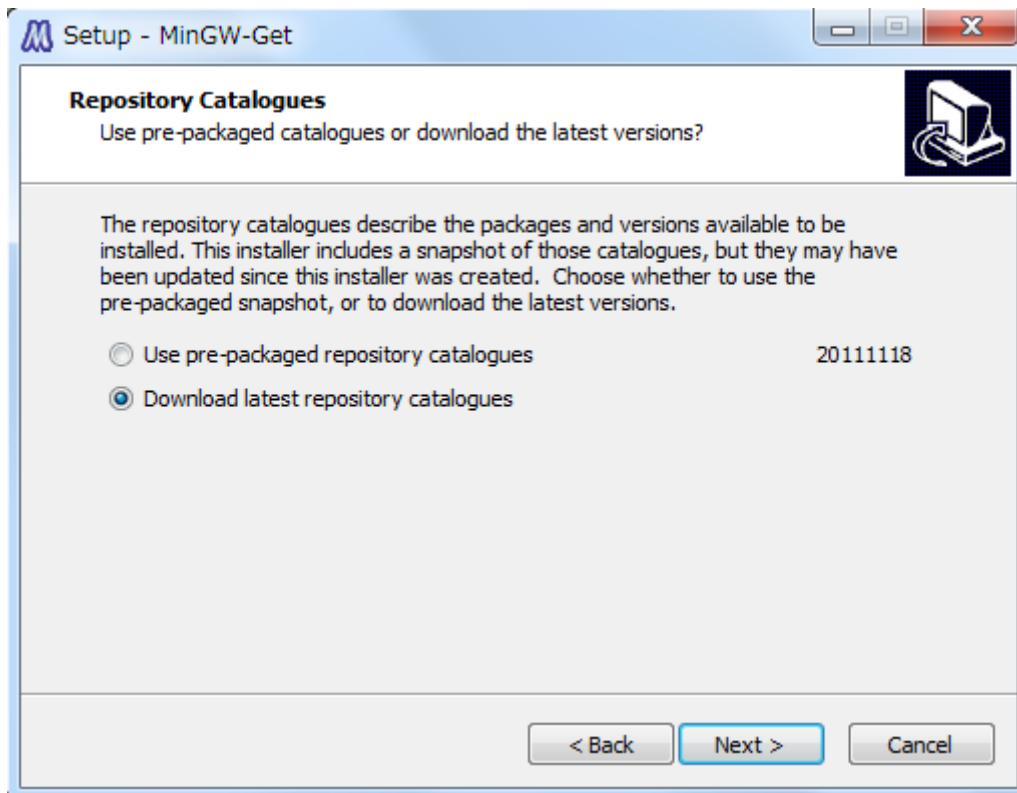


Step 3: As I am writing this in April 2012, the pre-packaged repository catalogs are a little dated. I advise you to instruct the installer to download the latest repository catalogs. And then proceed to click "next".
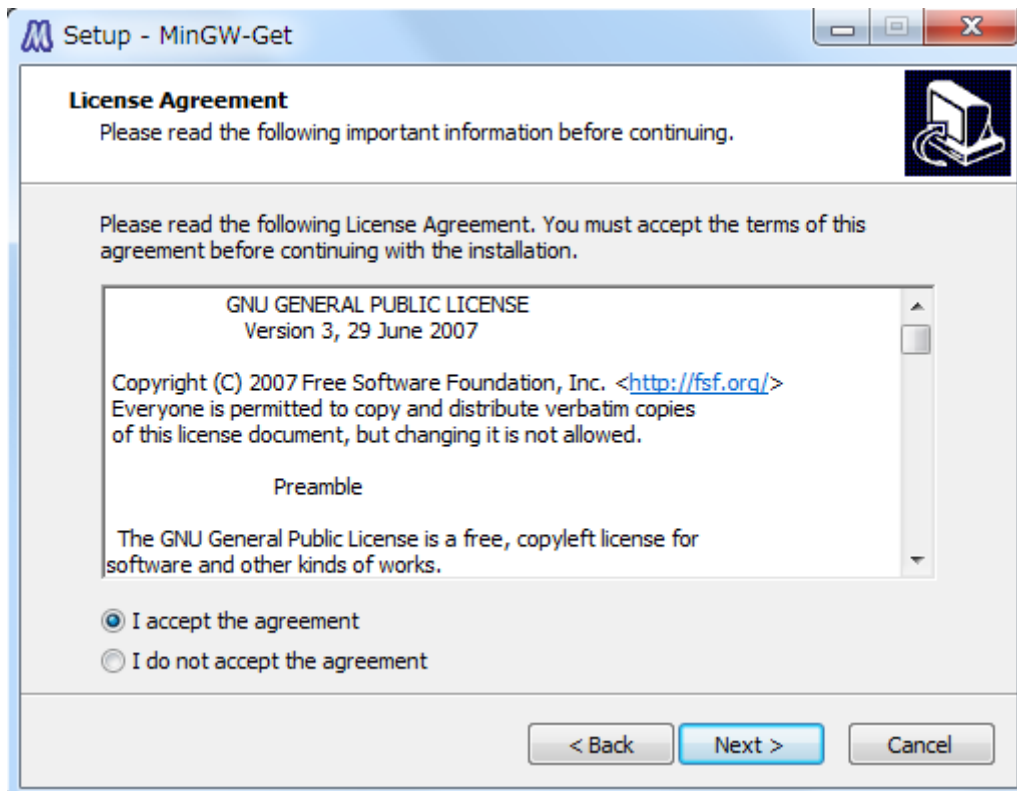
Step 4: After carefully reading the license agreement, either select "I agree" and click "next" or click "Cancel" if you do not accept the license agreement. If you do choose to click on the "Cancel" button, I would advise you to also stop reading and delete this tutorial, as it will no longer be beneficial to you. If you do click on "I agree" then you may proceed with this tutorial. This appears to be the same license agreement as Code::Blocks.
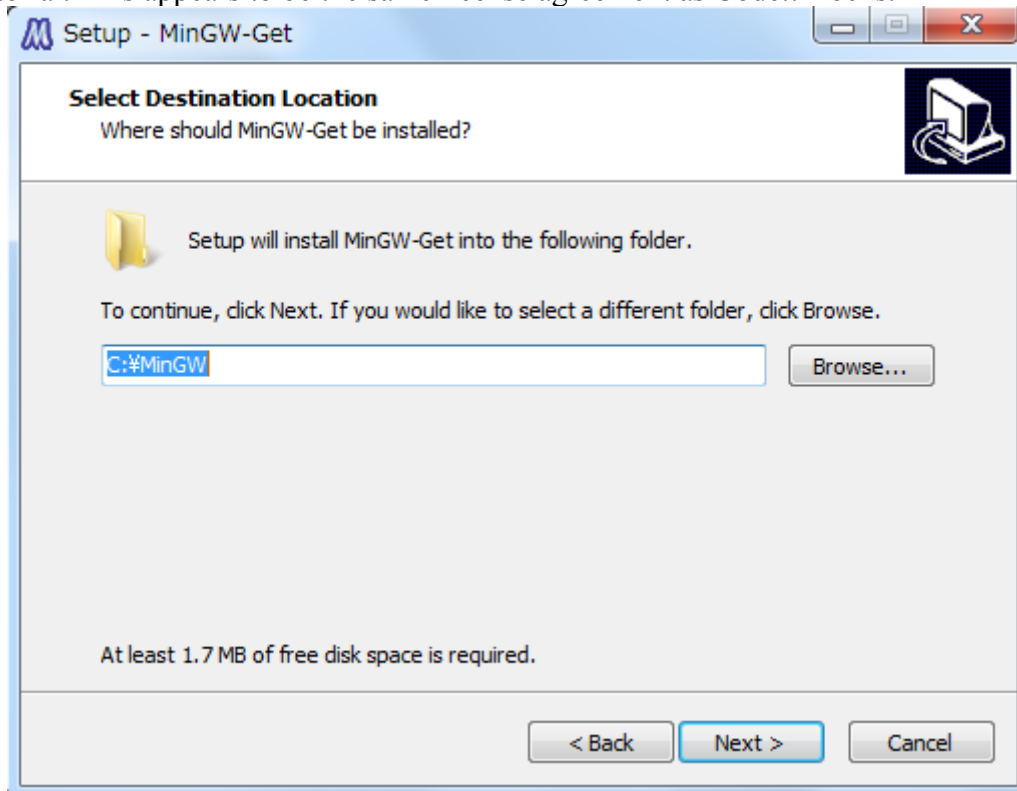


Step 5: On the next window we will be asked to specify the destination folder for our MinGW installation. You may change the default path, but for the purpose of this tutorial we will be installing MinGW into the default installation folder. So just click "next".
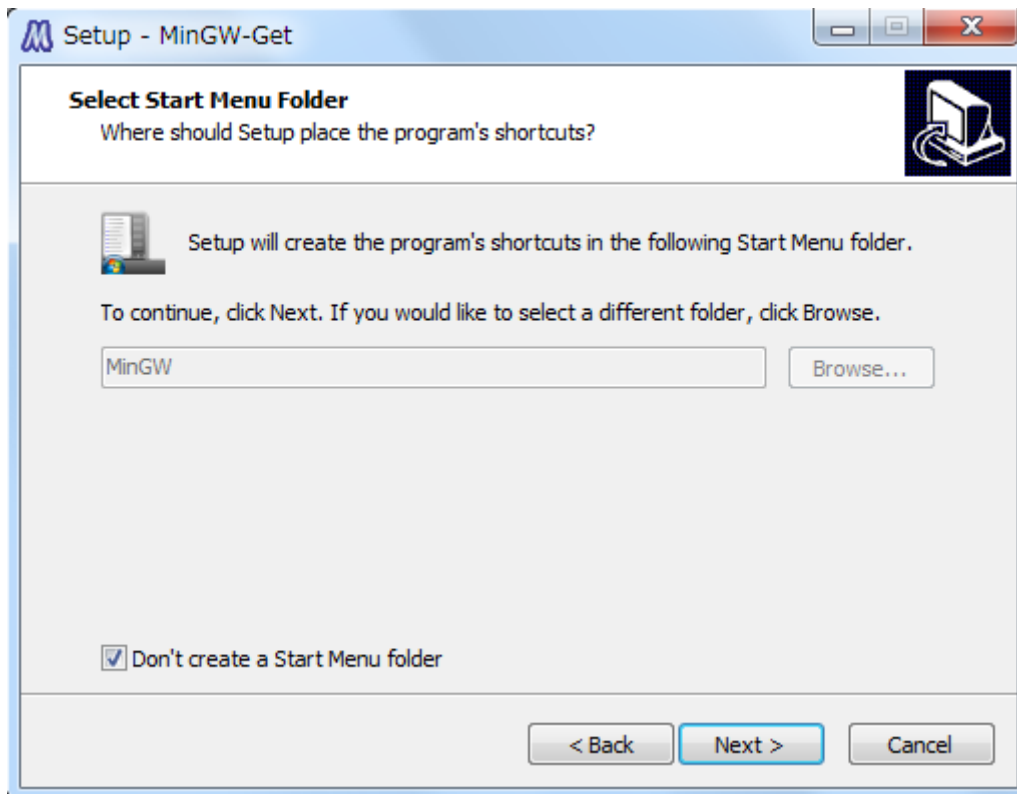
Step 6: You will probably never use the shortcuts in the start menu for MinGW, so you can just click "next". If you prefer to have the start menu shortcuts, feel free to change this, I doubt it will make any difference in what we are ultimately trying to achieve, the choice is yours.
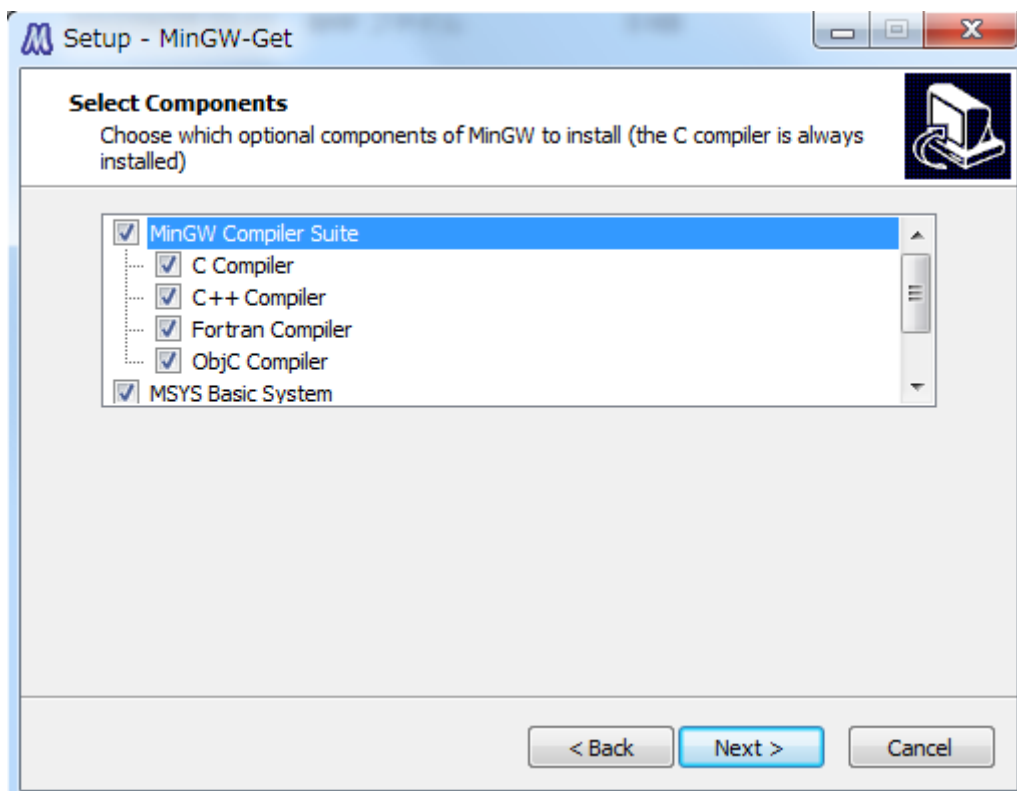


Step 7: Although we may not require all the features at this moment in time, we may however find them useful in the future, so we will be performing a full installation. I am aware of the fact that we don't need all these components for this tutorial, but for both simplicity and looking to future possibilities it makes more sence in the long term. The amount of disk space used by all the extra components is also negligible. Go ahead and click "next".
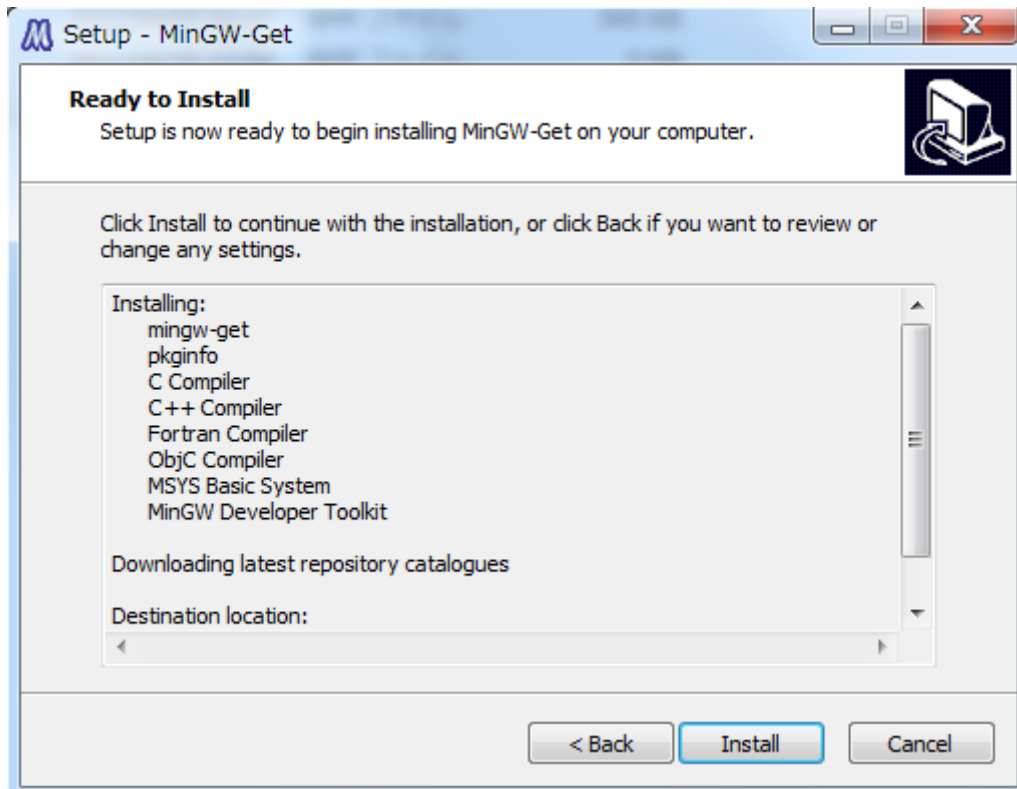
Step 8: You should be presented with a window such as this, click "Install". Now would be a good
time to get that cup of coffee/cola, it is not usually a long install, but coffee/cola is always a
good idea.
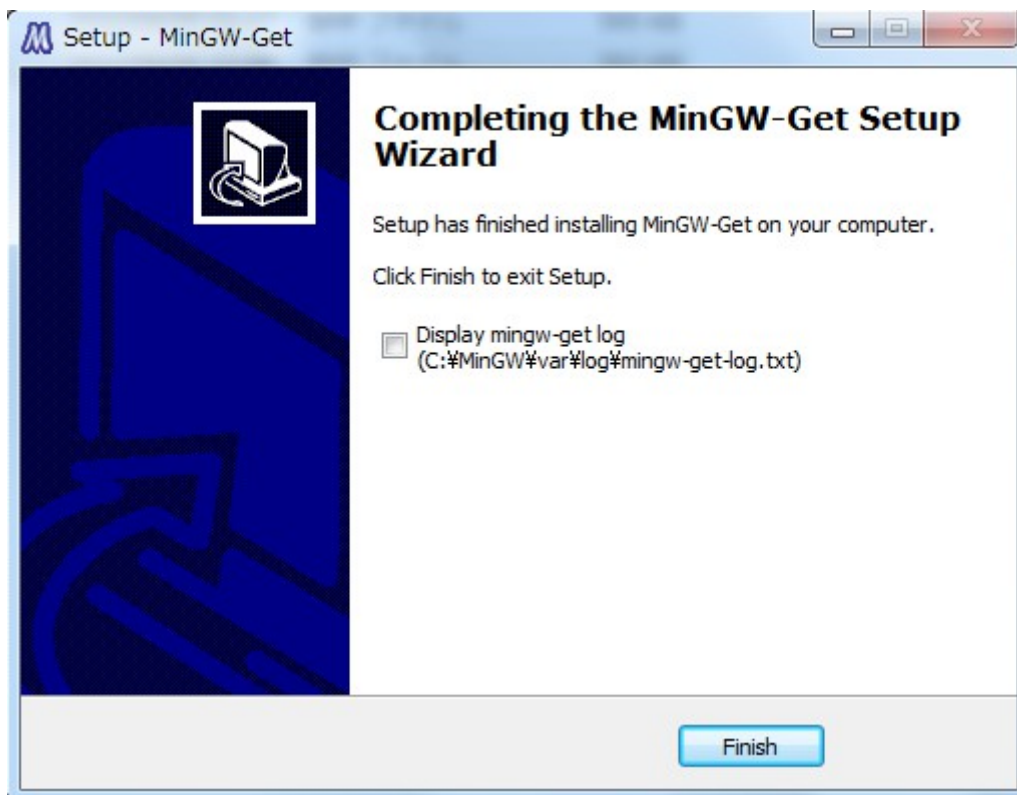


Step 9: Deselect "Display mingw-get log" at your discretion and click "finish".

# 4. Configuring Code::Blocks to use MinGW.

What we will be doing next is configuring Code::Blocks to use our MinGW compiler suite.
We will now proceed to run Code::Blocks for the first time.
If this is not your first time running Code::Blocks, don't worry, it should be fine to just move onto step 3, make sure GNU GCC is your default compiler in the settings-compiler settings though..
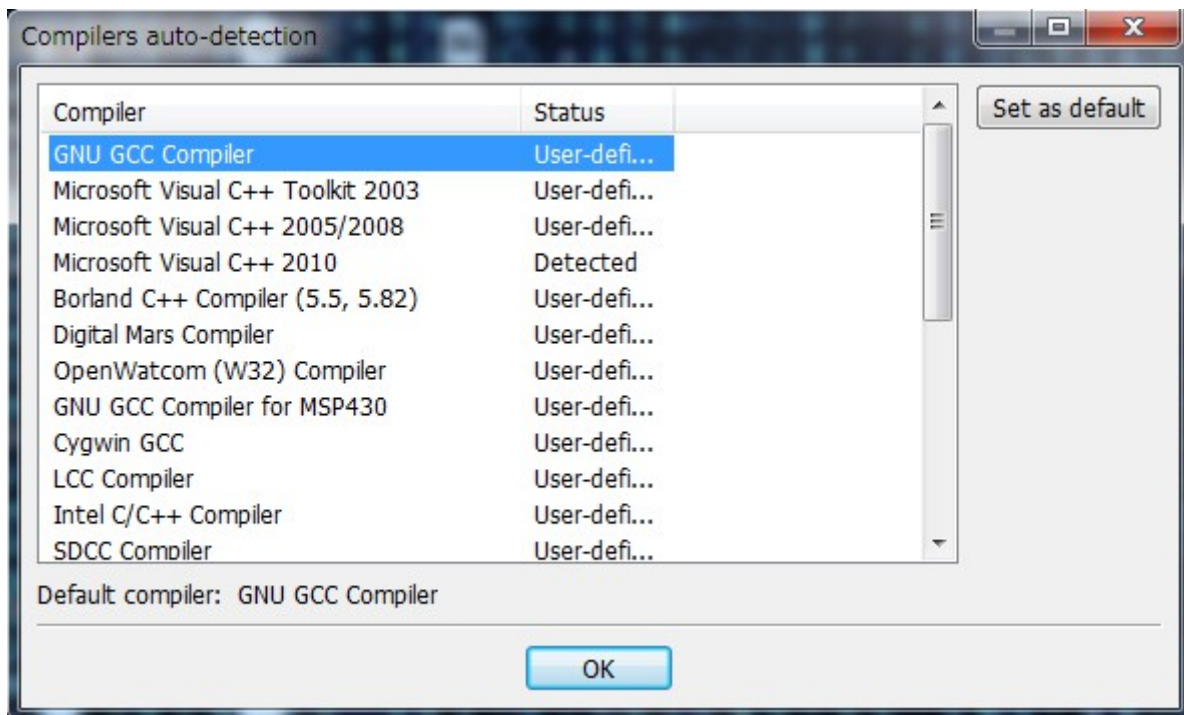


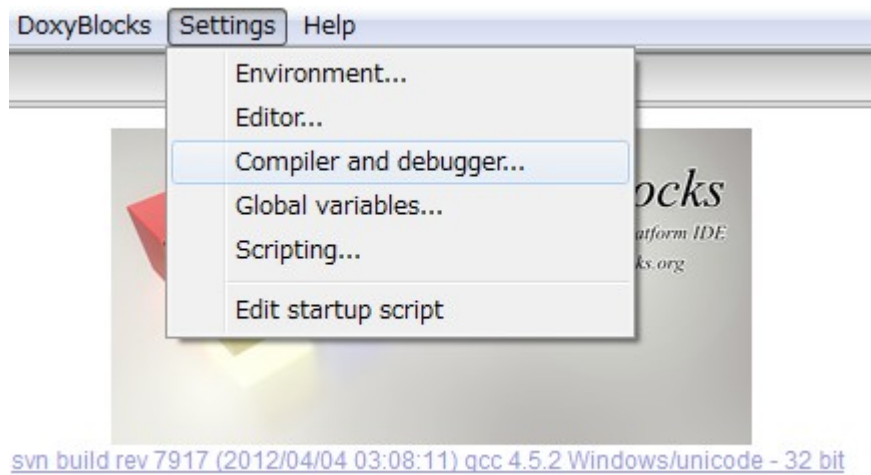Step 1: Select "GNU GCC Compiler" and then click on "Set as default", next continue by clicking "ok".
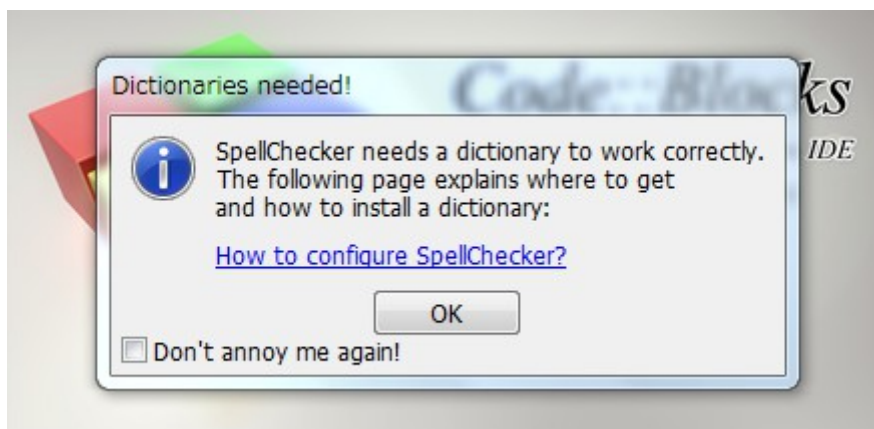


Step 2: If you installed the latest "nightly" build you will be prompted with this dialog box.
If you wish to use the SpellChecker plugin, you can find more information and guidance on that topic if you click on the "How to configure SpellChecker?" hyperlink.
As it is not vital, and just an added feature, also your language may be different from mine, and there is a link to information on how to configure it, this tutorial will not cover it.
I myself have installed a dictionary file and find it to be a nice feature.

Step 3: You will now find yourself inside the IDE, continue to click on "settings" on the top menu bar and the select "Compiler and debugger..."
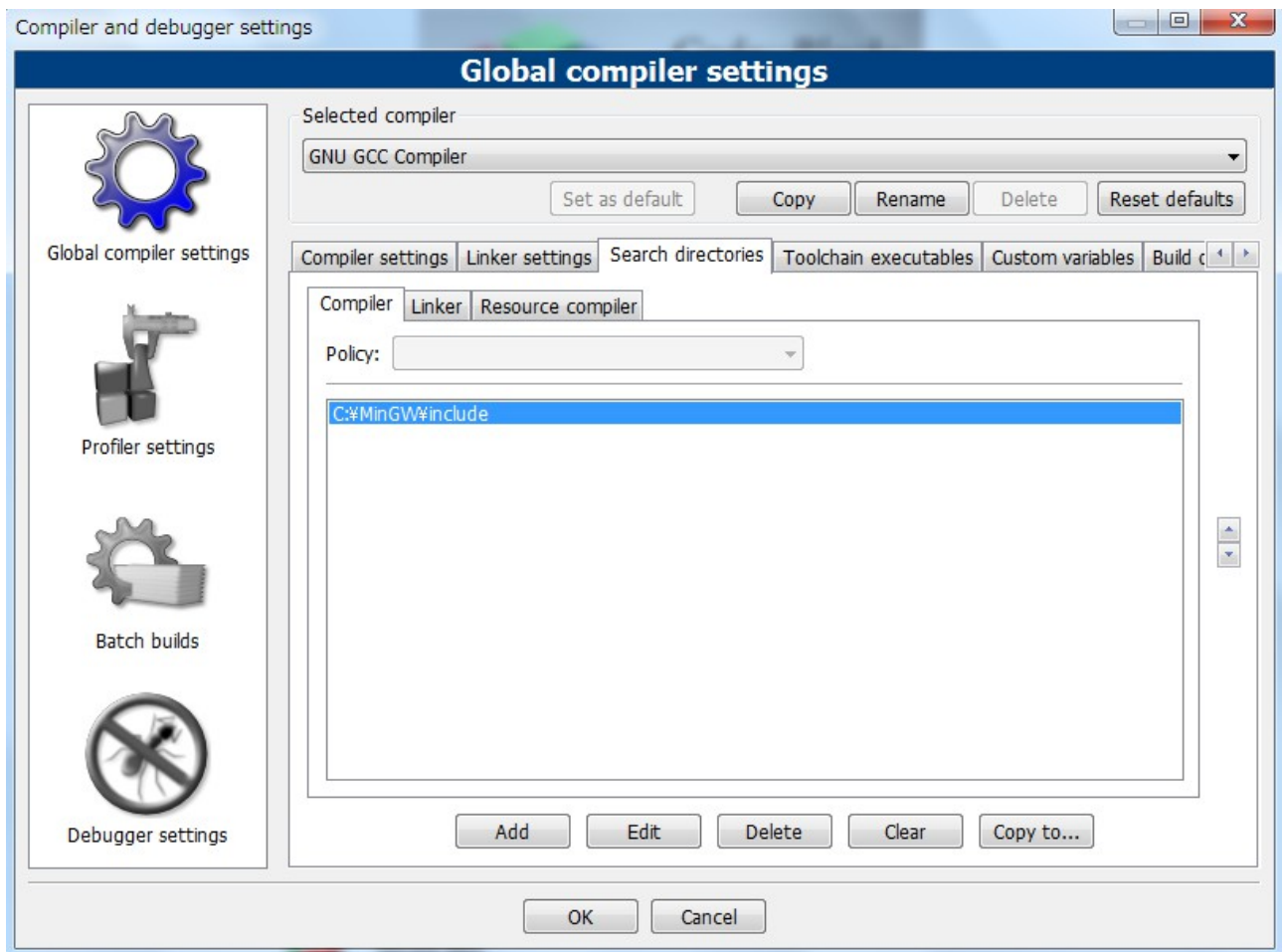


Step 4: We can now proceed to configure Code::Blocks with the GNU GCC Compiler, so if it is not selected as shown, select it from the drop-down menu and click on "Set as default".
Next click on "Search directories" and then select "Compiler". You will need to set the directory as shown, or if you installed MinGW elsewhere, set the directory accordingly.
In this tutorial we installed the MinGW compiler suite in the directory indicated in the picture. It should be "C:\MinGW\include".
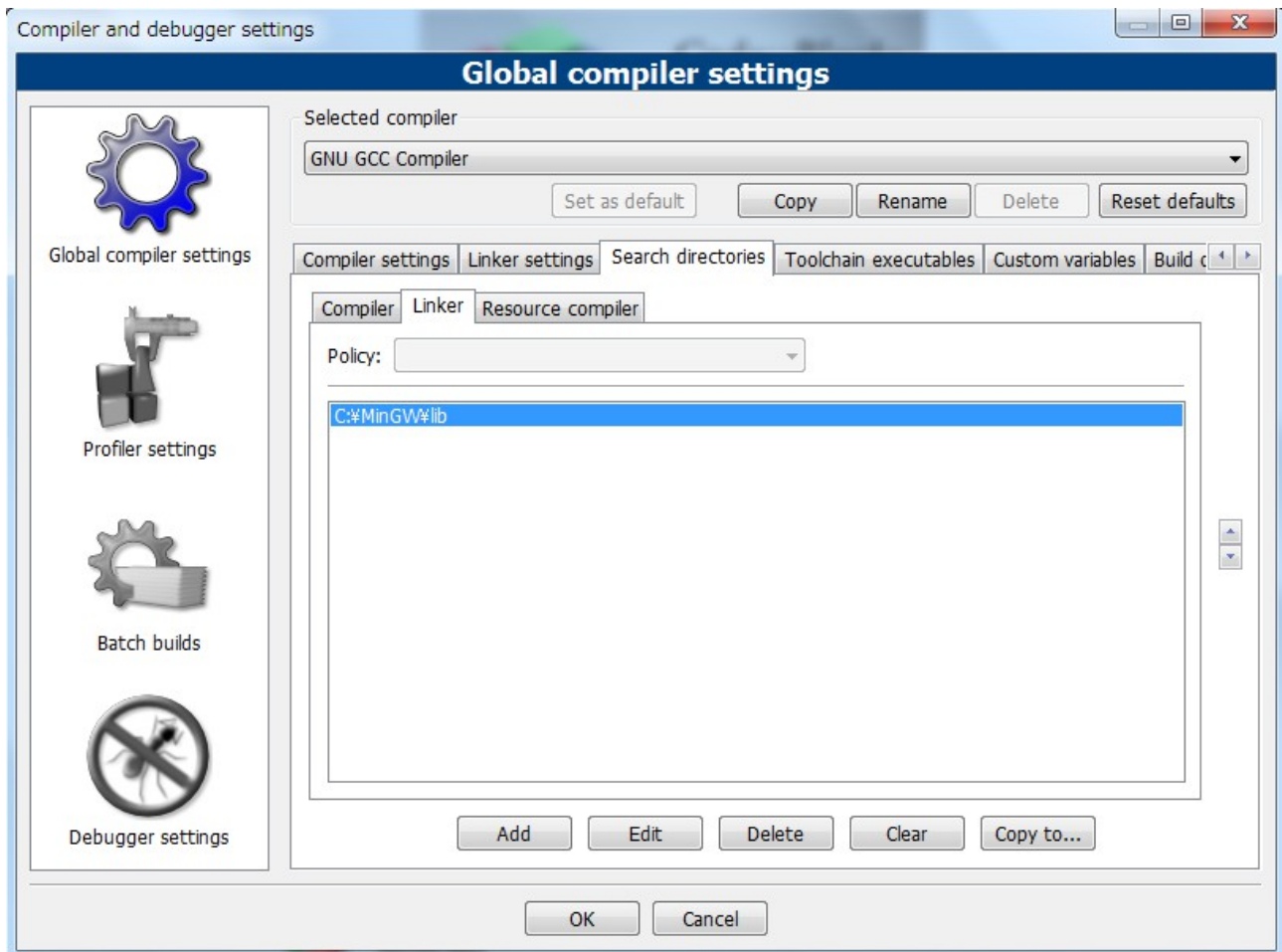
Step 5: Next click on "Linker" and define linker directory as shown. It should be "C:\MinGW\lib".
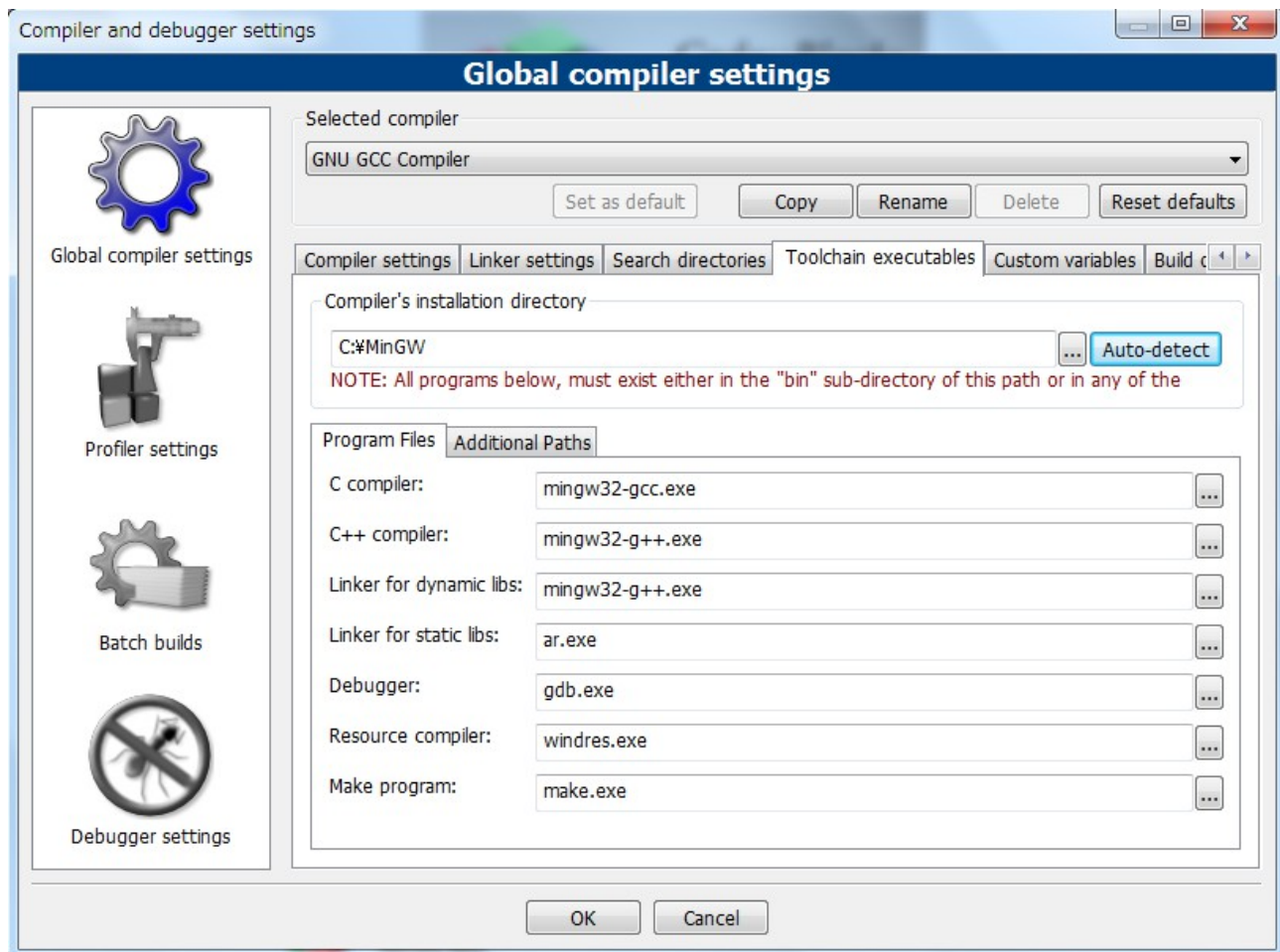
Step 6: We now have to configure the path/s to the toolchain executables. So click on "toolchain executables" and set the path as shown "C:\MinGW", or you can try the "Auto-detect" button and let Code::Blocks try find it for you. Under the "Program Files" tab, you should see all the file names as shown in the picture above.

# 5. Installing SDL, SDL_image, SDL_ttf and SDL_mixer.



## SDL

What we will be doing next is locating and installing the SDL API and some of its extendable components. Although there are other SDL components available, these three seem to be the most widely used and common, and for someone just starting out with SDL and computer programming we wont be installing any others in this tutorial. That said, if you are able successfully complete this tutorial, you should be able to install any other API that you require.
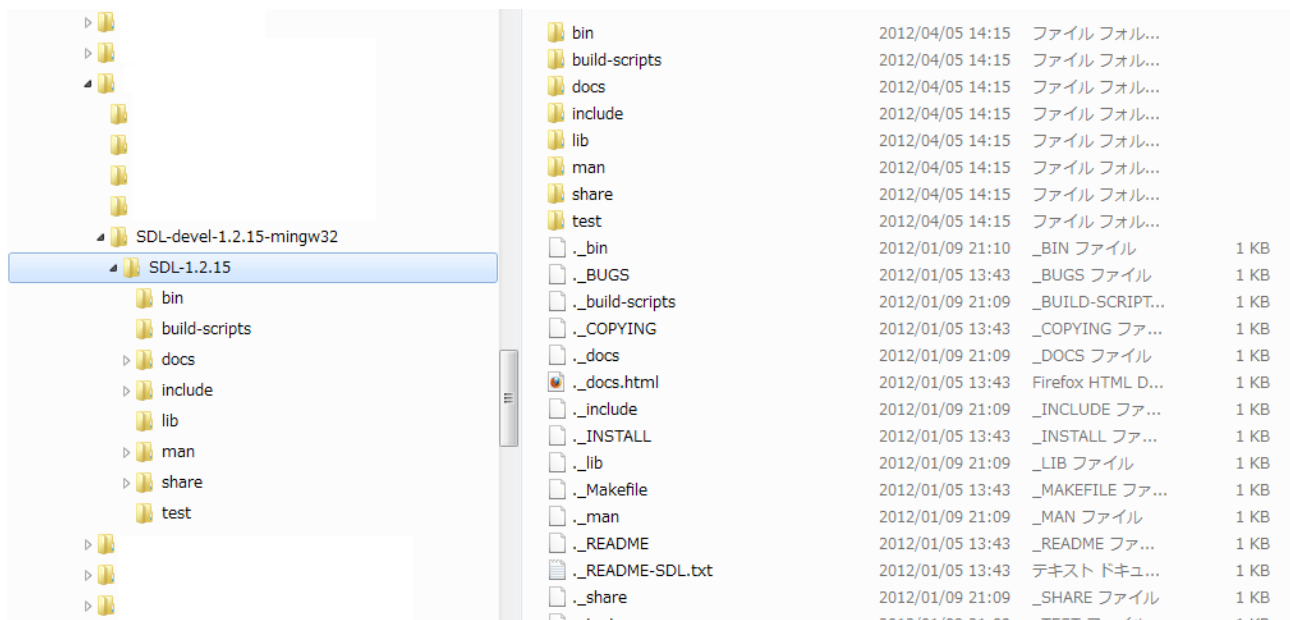The the SDL libraries webpage is http://www.libsdl.org/
What we will be installing is SDL version 1.2.15 (stable).
The download page is http://www.libsdl.org/download-1.2.php
Download the SDL development libraries for MinGW. SDL-devel-1.2.15-mingw32.tar.gz
Next decompress the archive and you should have a directory structured as below:



Next copy all the files and folders under the SDL-1.2.15 directory and paste them into your "C:\MinGW" folder. You will be replacing files, so just accept when windows prompts you.
Please note the directory structure in the SDL-1.2.15 folder and in the "C:\MinGW" folder.
You will be copying the files so as to keep the folder structure intact, ie. The "include" replaces the "include", the "lib" replaces the "lib" etc... (They actually merge)

**SDL_image**

Next we will be installing SDL_image  http://www.libsdl.org/projects/SDL_image/
Although there isn't a MinGW build available, we can use the VC one and it will should work just fine for most uses. It would be better to download the source and compile it with MinGW, but that is beyond the scope of this tutorial.
So as before we download the development library SDL_image-devel-1.2.12-VC.zip and decompress it into a temporary folder.
This time however we will be copying the files a little differently. Under the SDL_image-1.2.12 folder we have two folders, "include" and "lib".
We will copy the file "..\SDL_image-1.2.12 \include\SDL_image.h" from the temporary directory to the "C:\MinGW\include\SDL" directory.
After which we copy the "..\SDL_image-1.2.12\lib\x86\SDL_image.lib" file to the "C:\MinGW\lib" directory. Finally we copy all the remaining files (DLL's) from the "..\SDL_image-1.2.12\lib\x86\" directory to the "C:\MinGW\bin" directory.

**SDL_ttf**

The next library we will be installing is the SDL_ttf library, it can be found here:
http://www.libsdl.org/projects/SDL_ttf/

Again this time we will be downloading the VC development library. SDL_ttf-devel-2.0.11-VC.zip
Decompress it into a temporary folder, and copy the files into your MinGW folder as follows:

Under the SDL_ttf-2.0.11 folder we have two folders, "include" and "lib".
We will copy the file "..\SDL_ttf-2.0.11 \include\SDL_ttf.h" from the temporary directory to the "C:\MinGW\include\SDL" directory.
After which we copy the "..\SDL_ttf-2.0.11\lib\x86\SDL_ttf.lib" file to the "C:\MinGW\lib" directory. Finally we copy all the remaining files (DLL's) from the "..\SDL_ttf-2.0.11\lib\x86\" directory to the "C:\MinGW\bin" directory.

**SDL_mixer**

The next library we will be installing is the SDL_mixer library, it can be found here:
http://www.libsdl.org/projects/SDL_mixer/

Again this time we will be downloading the VC development library. SDL_mixer-devel-1.2.12-VC.zip Decompress it into a temporary folder, and copy the files into your MinGW folder as follows:

Under the SDL_mixer-1.2.12 folder we have two folders, "include" and "lib".
We will copy the file "..\SDL_mixer-1.2.12\include\SDL_mixer.h" from the temporary directory to the "C:\MinGW\include\SDL" directory.
After which we copy the "..\SDL_mixer-1.2.12\lib\x86\SDL_mixer.lib" file to the "C:\MinGW\lib" directory. Finally we copy all the remaining files (DLL's) from the "..\SDL_mixer-1.2.12\lib\x86\" directory to the "C:\MinGW\bin" directory.

Final notes:

To run an SDL application you will need to place the SDL runtime libraries (DLL files) in your compiled applications folder. These DLL files can be found in your "C:\MinGW\bin" folder.