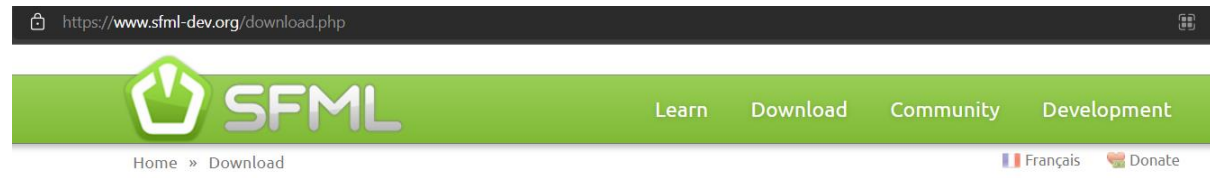
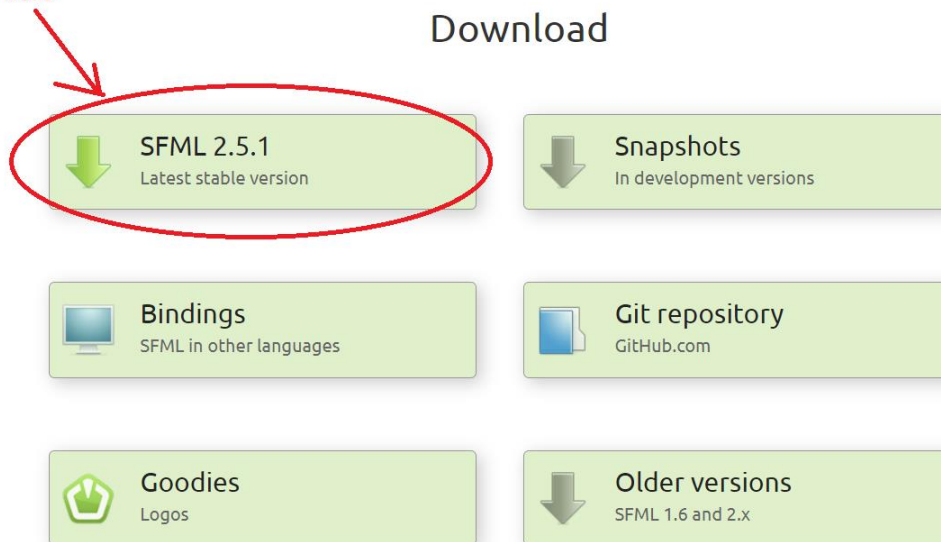


This is a step-by-step guide for building and using the software:

1: Download SFML and wingw32 from the SFML website. The download page URL is <https://www.sfml-dev.org/download.php>.



Click here



## Download SFML 2.5.1

On Windows, choosing 32 or 64-bit libraries should be based on which platform you want to compile for, not which OS you have. Indeed, you can perfectly compile and run a 32-bit program on a 64-bit Windows. So you'll most likely want to target 32-bit platforms, to have the largest possible audience. Choose 64-bit packages only if you have good reasons.

**The compiler versions have to match 100%!**

Here are links to the specific MinGW compiler versions used to build the provided packages:

TDM 5.1.0 (32-bit) | [MinGW Builds 7.3.0 \(32-bit\)](#) | [MinGW Builds 7.3.0 \(64-bit\)](#)

**wingw32 download**

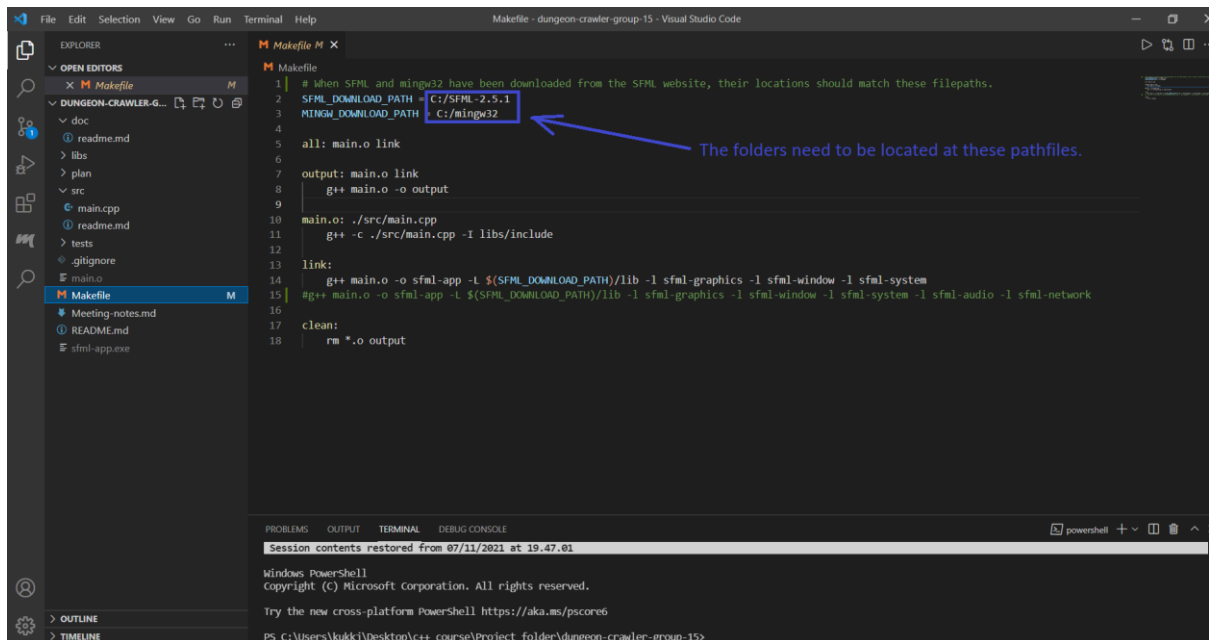
Visual C++ 15 (2017) - 32-bit	<a href="#">Download</a>   16.3 MB	Visual C++ 15 (2017) - 64-bit	<a href="#">Download</a>   18.0 MB
Visual C++ 14 (2015) - 32-bit	<a href="#">Download</a>   18.0 MB	Visual C++ 14 (2015) - 64-bit	<a href="#">Download</a>   19.9 MB
Visual C++ 12 (2013) - 32-bit	<a href="#">Download</a>   18.3 MB	Visual C++ 12 (2013) - 64-bit	<a href="#">Download</a>   20.3 MB
GCC 5.1.0 TDM (SJLJ) - Code::Blocks - 32-bit	<a href="#">Download</a>   14.1 MB		
GCC 7.3.0 MinGW (DW2) - 32-bit	<a href="#">Download</a>   15.5 MB	GCC 7.3.0 MinGW (SEH) - 64-bit	<a href="#">Download</a>   16.5 MB

**SFML-2.5.1 download**

On Linux, if you have a 64-bit OS then you have the 64-bit toolchain installed by default. Compiling for 32-bit is possible, but you have to install specific packages and/or use specific compiler options to do so. So downloading the 64-bit libraries is the easiest solution if you're on a 64-bit Linux. If you require a 32-bit build of SFML you'll have to [build it yourself](#).

It's recommended to use the SFML version from your package manager (if recent enough) or build from source to prevent incompatibilities.

2: After wingw32 and SFML-2.5.1 have been downloaded, extract the folders inside the downloaded WinRAR files and put the folders in a file location that matches the filepaths in the projects Makefile.

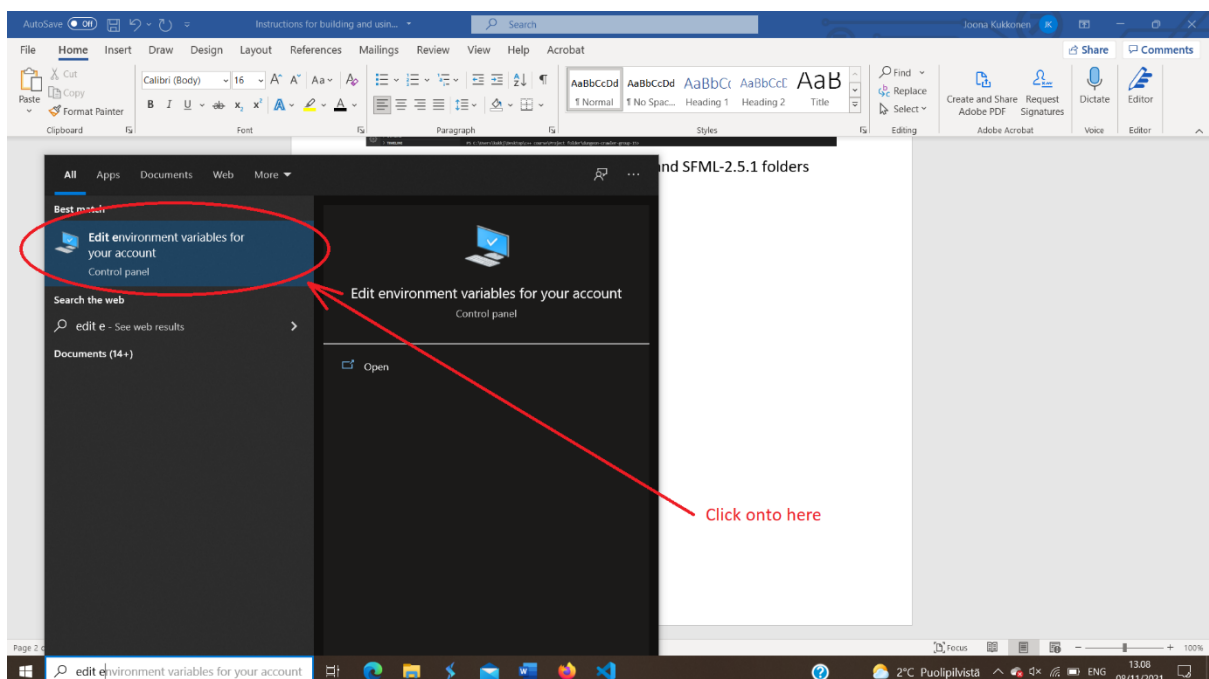


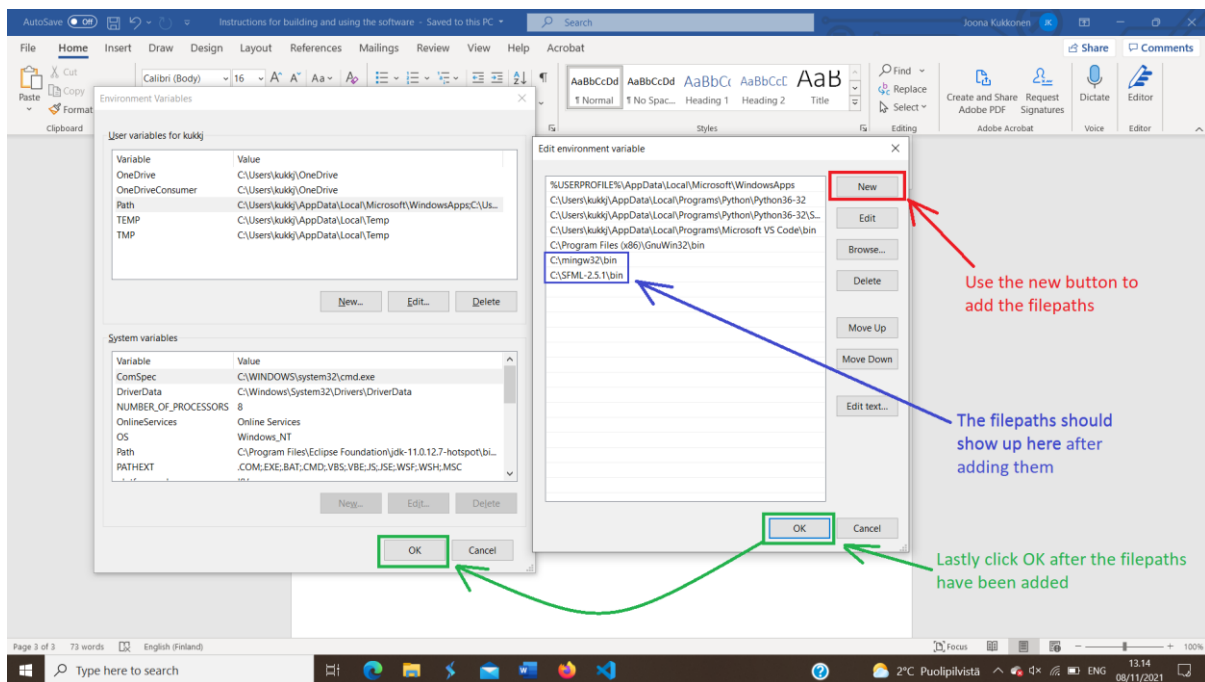
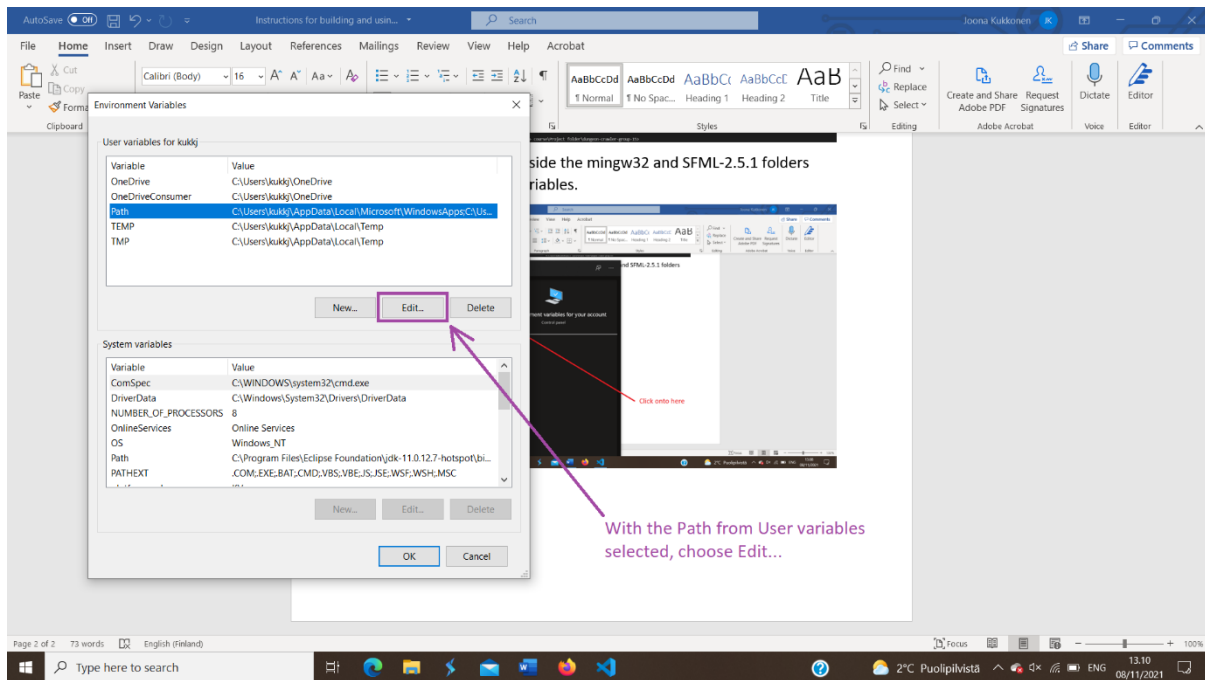
The screenshot shows the Visual Studio Code interface with a Makefile open. The Makefile contains the following content:

```
1 # When SFML and mingw32 have been downloaded from the SFML website, their locations should match these filepaths.
2 SFML_DOWNLOAD_PATH = C:/SFML-2.5.1
3 MINGW_DOWNLOAD_PATH = C:/mingw32
4
5 all: main.o link
6
7 output: main.o link
8 g++ main.o -o output
9
10 main.o: ./src/main.cpp
11 g++ -c ./src/main.cpp -I libs/include
12
13 link:
14 g++ main.o -o sfml-app -l $(SFML_DOWNLOAD_PATH)/lib -l sfml-graphics -l sfml-window -l sfml-system
15 #g++ main.o -o sfml-app -l $(SFML_DOWNLOAD_PATH)/lib -l sfml-graphics -l sfml-window -l sfml-system -l sfml-audio -l sfml-network
16
17 clean:
18 rm *.o output
```

A blue arrow points to the paths in the Makefile with the text: "The folders need to be located at these filepaths."

3: Add the bin folders inside the mingw32 and SFML-2.5.1 folders into environment file variables.





4: Your SFML is now set up properly. You now need to download make in order to run the Makefile. After make has been downloaded, go to the projects terminal window and write the command make.

