

Anyone Can Code

The Art and Science of Logical Creativity

Software Guide

Introduction

This document briefly describes how to use the sample codes in the book *Anyone Can Code: The Art and Science of Logical Creativity*. The samples are given in three different languages: Javascript, Python, and C/C++. C and C++ are two related but separate languages. C++ is an extension to C that introduces object-oriented programming and classes. Earlier chapters of the book are compatible with both C and C++, while the chapters after object and classes are introduced use features unique to C++.

The sample for Python and C/C++ also use external libraries for graphics that are not standard in the languages.

The following chapters review what is needed for all the above cases.

Javascript

Javascript samples use standard libraries. Almost all Javascript samples are embedded within an HTML file.

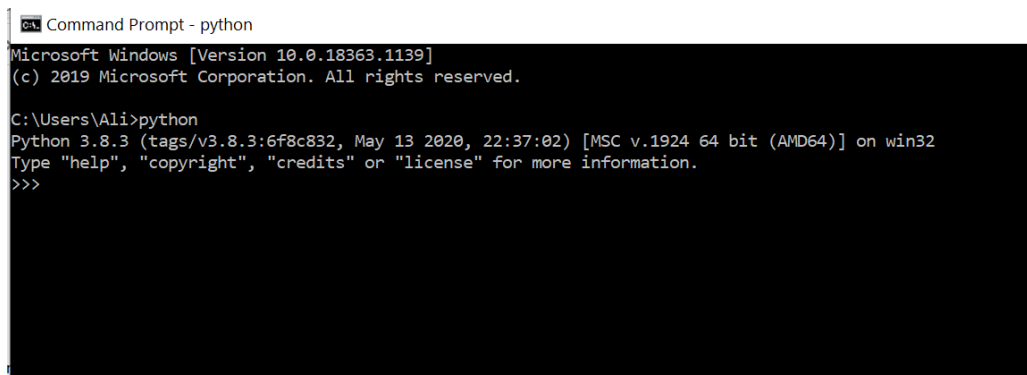
To view and edit the code, you can open them with any text editor or a specialized development environment such as Adobe DreamWeaver or Microsoft Visual Studio.

To run the code, open the HTML file with any web browser that has Javascript support. The graphics samples require HTML-5 support. If the sample has multiple files, make sure they are all copied to the same folder on your computer.

Python

Python examples require Python to be installed on your computer. You can download and install Python using the instructions here: <https://www.python.org/doc/versions/>

Before trying the sample codes, make sure Python is properly installed. Open a command prompt (terminal) and type **py** (or **python**) and you should see something like this that shows the version number:



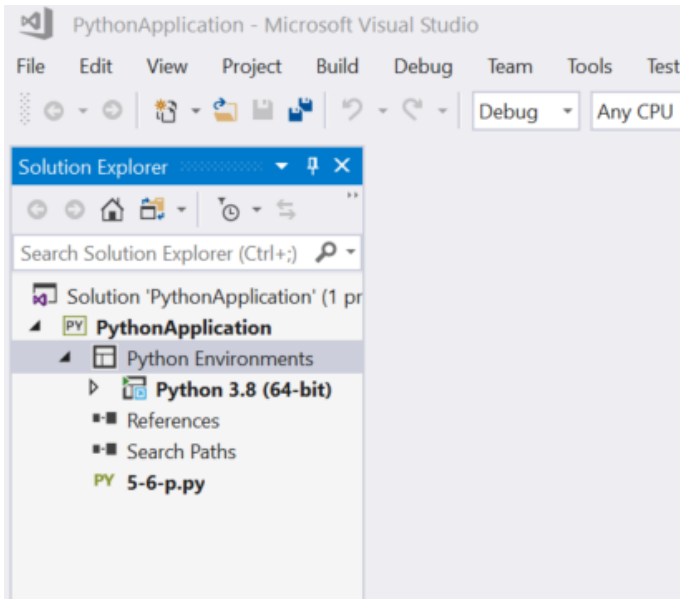
```
Command Prompt - python
Microsoft Windows [Version 10.0.18363.1139]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Ali>python
Python 3.8.3 (tags/v3.8.3:6f8c832, May 13 2020, 22:37:02) [MSC v.1924 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

You can view and edit the sample code by opening them with a text editor or a specialized development environment like Visual Studio.

To run the samples, you may open a command prompt (terminal), go to the folder where your source files are, and type “py filename.py” where filename.py is the name of your Python file. If you have multiple source files, one of them should be your starting file and you use its name. That file will use the others as long as all files are in the same folder.

Alternatively, you can create a Python project in your development tool, and add all source files to it, and choose the starting one. A Visual Studio project is provided as a base that you can use to add files to. You may need to choose the right version of Python if you have multiple installed or you have a different version from what is defined in the sample project. In Visual Studio, look for Python Environments under the project name.



To add a source file to the project, right-click on the project name and choose Add Existing.

To remove a file from the project, right-click on the file and choose Exclude from the Project.

To choose the start-up file, make sure you right-click on the file and choose Set as Start-up.

Graphics samples use graphics.py library. It has to be included in the project and be in the same folder as your source files.

C/C++

The C/C++ samples in the book are three types: console application (text-only), graphics using SDL library, and graphics using OpenFrameworks. While you can view and edit the C/C++ files with a text editor and use a command-line or online compiler, it is not recommended to do so. I suggest using Visual Studio or a similar development environment.

The C/C++ applications should be compatible with most platforms as long as the library they are using is available and installed.

Console Applications

Console applications use standard C/C++ libraries and are text-only. Create a new empty Console Application in Visual Studio and add the source files by right-clicking on the project name and using Add Existing. I recommend copying the source files to your project folder first. For multiple file projects, you need to add them all.

The console samples use a void type for main() function that may not be supported in some compilers. If that happens, change the type from void to int and add "return 0;" to the end of main().

Run the program from within Visual Studio.

SDL/SDLX

SDL is graphics library for C and is compatible with many platforms. You can download and install it here:

<https://www.libsdl.org/>

SDL is written without the use of classes so can be use din both C and C++ programs (with or without classes).

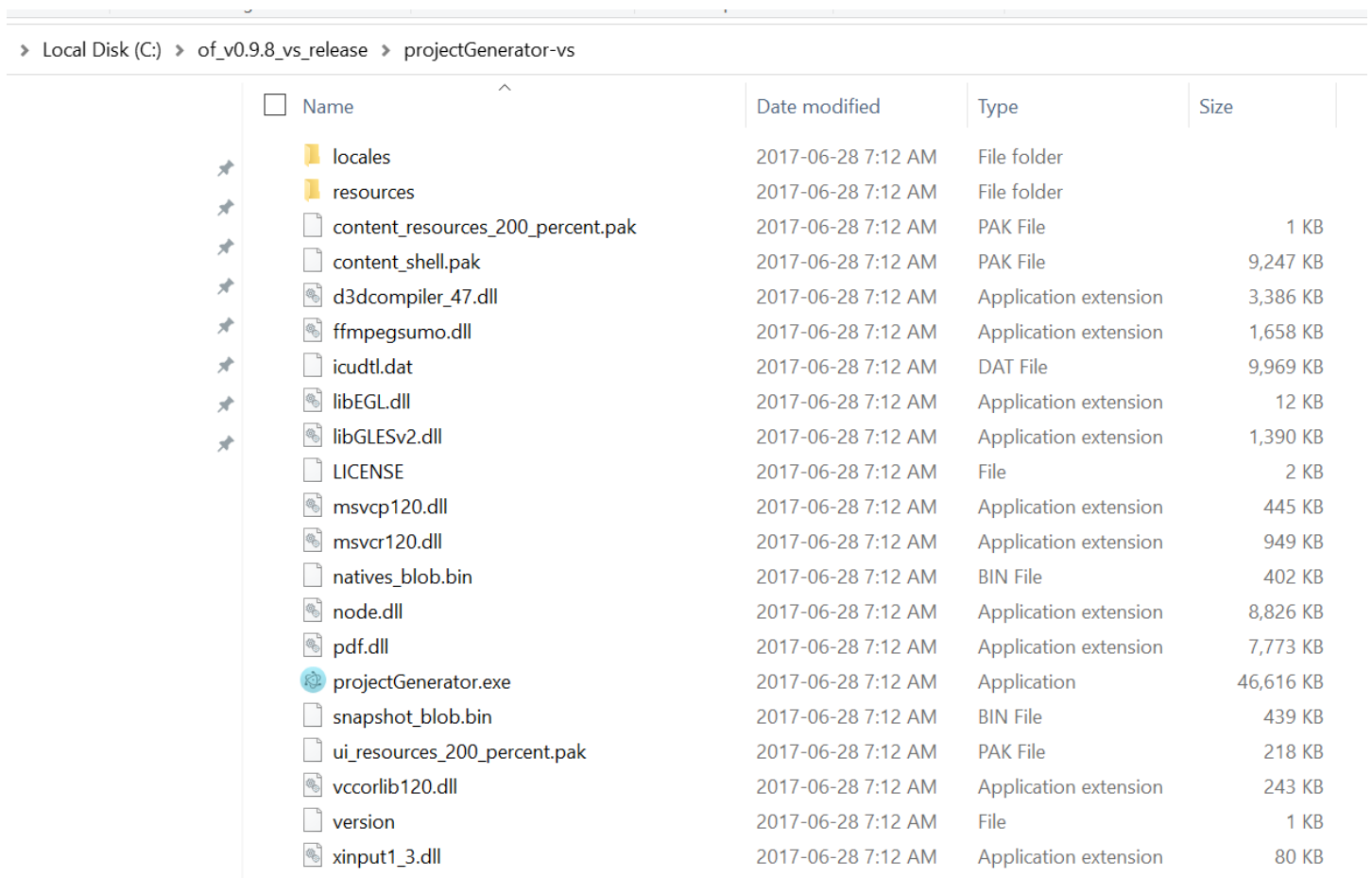
SDLX is an extra layer on top of SDL for more convenience. **SDLX.h** and **SDLX.cpp** have to be included in all sample SDL projects.

To use the SDL samples, create an empty Windows Desktop application and add the source files to it. As before, I recommend you copy all the source and image files to the project folder first. I also recommend you use the sample SDL project (provided with the book sample codes), instead of creating new projects.

OpenFrameworks

OpenFrameworks (OF) is a graphics library for C++, so I has to be used in an object-oriented program. You can download and install OF here: <https://openframeworks.cc/>

To use OF samples, create a new project using the **projectGenerator** tool that comes with OF:



Name	Date modified	Type	Size
locales	2017-06-28 7:12 AM	File folder	
resources	2017-06-28 7:12 AM	File folder	
content_resources_200_percent.pak	2017-06-28 7:12 AM	PAK File	1 KB
content_shell.pak	2017-06-28 7:12 AM	PAK File	9,247 KB
d3dcompiler_47.dll	2017-06-28 7:12 AM	Application extension	3,386 KB
ffmpegsumo.dll	2017-06-28 7:12 AM	Application extension	1,658 KB
icudtl.dat	2017-06-28 7:12 AM	DAT File	9,969 KB
libEGL.dll	2017-06-28 7:12 AM	Application extension	12 KB
libGLv2.dll	2017-06-28 7:12 AM	Application extension	1,390 KB
LICENSE	2017-06-28 7:12 AM	File	2 KB
msvcp120.dll	2017-06-28 7:12 AM	Application extension	445 KB
msucr120.dll	2017-06-28 7:12 AM	Application extension	949 KB
natives_blob.bin	2017-06-28 7:12 AM	BIN File	402 KB
node.dll	2017-06-28 7:12 AM	Application extension	8,826 KB
pdf.dll	2017-06-28 7:12 AM	Application extension	7,773 KB
projectGenerator.exe	2017-06-28 7:12 AM	Application	46,616 KB
snapshot_blob.bin	2017-06-28 7:12 AM	BIN File	439 KB
ui_resources_200_percent.pak	2017-06-28 7:12 AM	PAK File	218 KB
vccorlib120.dll	2017-06-28 7:12 AM	Application extension	243 KB
version	2017-06-28 7:12 AM	File	1 KB
xinput1_3.dll	2017-06-28 7:12 AM	Application extension	80 KB

Alternatively, you can use the sample OF project that is provided with book sample codes. Make sure that the project is copied to the right location:

Local Disk (C:) > of_v0.9.8_vs_release > apps > myApps

<input type="checkbox"/>	Name	Date modified	Type	Size
✦	addons	2019-04-25 5:48 PM	File folder	
✦	MyOFProject	2020-09-07 3:51 PM	File folder	
✦	OF	2020-11-05 10:50 AM	File folder	
✦	OF-AnimatedObject	2020-10-08 10:22 AM	File folder	
✦	<input type="checkbox"/> OF-AnimatedObject-v1	2020-09-13 4:10 PM	File folder	
✦	OF-AnimatedObject-v2	2020-09-13 4:15 PM	File folder	
✦	OF-Game-A3	2020-10-10 11:32 AM	File folder	
✦	OF-Game-C	2020-02-27 4:58 PM	File folder	
✦	OF-GameObject-v1	2020-10-07 1:36 PM	File folder	
✦	OF-GameObject-v2	2020-09-13 4:05 PM	File folder	
✦	OF-Game-v3	2020-10-08 9:29 AM	File folder	
✦	OF-Game-v4	2020-10-26 1:50 PM	File folder	
✦	OF-LinkedList	2020-10-22 9:45 AM	File folder	
✦	OF-Base.zip	2020-10-31 5:19 PM	Compressed (zipped)...	5,365 KB

OF source files are in a folder named src under the project folder. The data files such as images are under bin/data. Each OF sample comes with two folders: src for code and data for images. Copy the content of each folder to the corresponding folder in your project. Then, in Visual Studio, **right click on the src under your project name**, and choose Add Existing.