

Занятие 2

Stake наиболее эффективный инструмент при работе со сторонними расширениями, пакетами, библиотеками.

В этой работе будет рассмотрено подключение сторонней библиотеки и компиляция проекта.

Для начала разберемся, что из себя представляет библиотека.

Библиотека – это сборник подпрограмм или объектов, используемых для разработки программного обеспечения. Иными словами, библиотеки – это уже готовый код, который программист может использовать в своих проектах.

С точки зрения операционной системы и прикладного ПО, библиотеки разделяются на динамические и статические.

Статические библиотеки представляют собой скомпилированный код, который встраивается в наш исполняемый файл в процессе линковки программы.

Динамические библиотеки представляют собой отдельные компоненты кода, как правило в виде отдельных файлов, поставляемых вместе с исполняемой программой. Загрузка динамической библиотеки в память происходит в процессе работы программы.

Распространённые форматы статических библиотек: .lib, .a.

Распространённые форматы динамических библиотек: .dll, .so, .dylib.

Когда мы разобрались с принципом работы библиотек и их видами, давайте попробуем подключить реальную библиотеку в наш проект на примере SDL2.

Задание

1. Скачайте библиотеку инструменты разработки SDL2-devel, а именно заголовочные файлы и библиотеку, поищите инструкцию по установке в интернете. Этот ресурс может быть полезен [Simple DirectMedia Layer - SDL version 2.0.14 \(stable\) \(libsdl.org\)](https://www.libsdl.org/release/SDL2-devel-2.0.14.tar.gz).
2. Основной код программы выглядит следующим образом, запишите его в файл с расширением .cpp или .c.

```
#include <SDL2/SDL.h>
const int SCREEN_WIDTH = 640;
const int SCREEN_HEIGHT = 480;

int main (int argc, char ** args) {

    if( SDL_Init( SDL_INIT_EVERYTHING ) != 0 )
    {
        return 1;
    }
}
```

```

SDL_Surface* screen_surface = NULL;

SDL_Window* window = NULL;

window = SDL_CreateWindow("Hello, SDL 2!",SDL_WINDOWPOS_UNDEFINED,
SDL_WINDOWPOS_UNDEFINED, SCREEN_WIDTH, SCREEN_HEIGHT,
SDL_WINDOW_SHOWN);

if (window == NULL) {
    return 1;
}

screen_surface = SDL_GetWindowSurface(window);

SDL_FillRect(screen_surface, NULL,
             SDL_MapRGB( screen_surface->format, 0, 255, 0));

SDL_UpdateWindowSurface(window);

SDL_Delay(2000);

SDL_DestroyWindow(window);

SDL_Quit();

return 0;
};

```

3. Приступим к заполнению CMakeLists.txt.

```

cmake_minimum_required(VERSION 3.17)

project(lesson2)

add_executable(lesson2 test.cpp)

```

Для того чтобы указать в нашем проекте используемые библиотеки можно воспользоваться опцией `target_link_libraries`.

Эта опция ищет файлы наших библиотек в определенных директориях и линкует их с программой, если библиотека там есть.

Для MinGW под Windows добавьте строчку в CMakeLists.txt.

```
target_link_libraries(lesson2 mingw32 SDL2main SDL2)
```

Мы указали библиотеки `mingw32` `SDL2main` и `SDL2` в качестве параметров, библиотека `mingw32` нужна для правильной компиляции MinGW и может не применяться с другими компиляторами.

4. Произведем сборку последовательно выполнив команды

```
C:\Users\grish\Documents\exp\SDL\lesson 2\build>cmake .. -G"MinGW Makefiles"
```

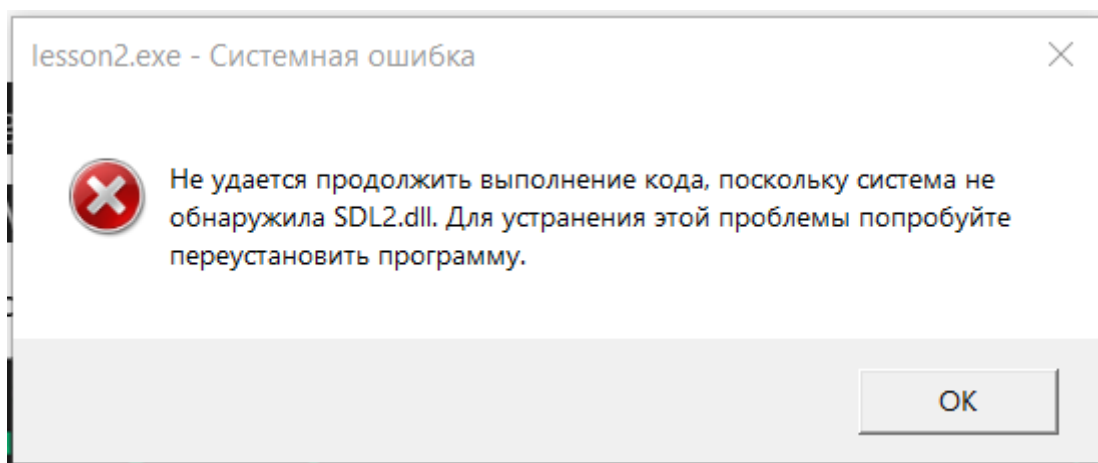
```
C:\Users\grish\Documents\exp\SDL\lesson 2\build>cmake --build .
```

Сборка прошла успешно

```
Scanning dependencies of target lesson2
[ 50%] Building CXX object CMakeFiles/lesson2.dir/test.cpp.obj
[100%] Linking CXX executable lesson2.exe
[100%] Built target lesson2
```

5. Запустите программу

Если вы запустили программу и увидели следующую ошибку, это значит, что при загрузке динамической библиотеки наша программа не смогла её найти.



Есть несколько способов решения данной проблемы: добавить dll файл в системную директорию, куда Windows обращается для поиска сторонних библиотек; добавить путь к dll файлу в system path; положить dll рядом с exe файлом.

Мы воспользуемся последним вариантом (если у вас нет SDL2.dll, его можно найти в интернете) и добавим файл в каталог, где находится lesson2.exe.

CMaFiles	06.06.2021 20:30	Папка с файлами	
cmake_install.cmake	06.06.2021 20:26	Файл "CMAKE"	2 КБ
CMakeCache.txt	06.06.2021 20:26	Текстовый докум...	16 КБ
lesson2.exe	06.06.2021 20:30	Приложение	58 КБ
Makefile	06.06.2021 20:30	Файл	6 КБ
SDL2.dll	28.10.2020 17:47	Расширение при...	5 649 КБ

Если все было сделано правильно мы получим следующий результат:

