Task 15 - SDL2 Concepts Lab Notes
104001272 - Hayden Whiteford
Games Programming

**Questions**

1. https://www.libsdl.org/ is the official URL for the SDL2 website
2. SDL stands for Simple DirectMedia Layer
3. It is available under the zlib License.
4. SDL officially supports Windows, macOS, Linux, iOS, and Android.
5. It is written in C.
6. https://wiki.libsdl.org/SDL2/CategoryAPI allows for searching through documentation by Name.
7. Downloading SDL2 from source code provides you with the entirety of the SDL library source code. Its really only necessary if you're aiming to develop or modify SDL2 yourself. Downloading the Runtime Binaries give you only the precompiled binaries required to run SDL based applications, which is perfect for users. The Development libraries include the header files and other important files that are required for the development of applications that will use SDL.
8. The Development Libraries would be the option I'd want to use, so I can use SDL2 in my c++ projects in Xcode
9. The difference here is that SDL2 is only providing low level access to hardware such as audio, keyboard and mouse, and other hardware. A game framework would be building upon some of the foundation of SDL Libraries or similar ones in order to create a suitable environment for game development.

**Simple SDL Test Code**

```
#include <SDL2/SDL.h>
#include <cstdlib>
#include <ctime>

int main(int argc, char* argv[]) {

    // Create a window
    SDL_Window* window = SDL_CreateWindow("SDL2 Demo", 
SDL_WINDOWPOS_UNDEFINED, SDL_WINDOWPOS_UNDEFINED, 800, 600, 
SDL_WINDOW_SHOWN);

    // Create a renderer
    SDL_Renderer* renderer = SDL_CreateRenderer(window, -1, 
SDL_RENDERER_ACCELERATED);

    // Initialize random seed
    std::srand(static_cast<unsigned int>(std::time(nullptr)));

    bool quit = false;
    SDL_Event event;
    SDL_Color backgroundColor = {0, 255, 0, 255}; // Initial 
background color: Green

    while (!quit) {
        while (SDL_PollEvent(&event) != 0) {
            //only ending the loop when the SDL_QUIT event is 
triggered by closing the window
```

```
            if (event.type == SDL_QUIT) {
                quit = true;
            }
            else if (event.type == SDL_KEYDOWN) {
                if (event.key.keysym.sym == SDLK_r) {
                    // Change background color to a random color
                    backgroundColor.r = std::rand() % 256;
                    backgroundColor.g = std::rand() % 256;
                    backgroundColor.b = std::rand() % 256;
                }
            }
        }

        // Set the background color
        SDL_SetRenderDrawColor(renderer, backgroundColor.r,
backgroundColor.g, backgroundColor.b, backgroundColor.a);
        SDL_RenderClear(renderer);

        // Present the renderer
        SDL_RenderPresent(renderer);
    }

    // Cleanup and quit
    SDL_DestroyRenderer(renderer);
    SDL_DestroyWindow(window);
    SDL_Quit();
    return 0;
}
```