DAY3 SDL to Create a Window

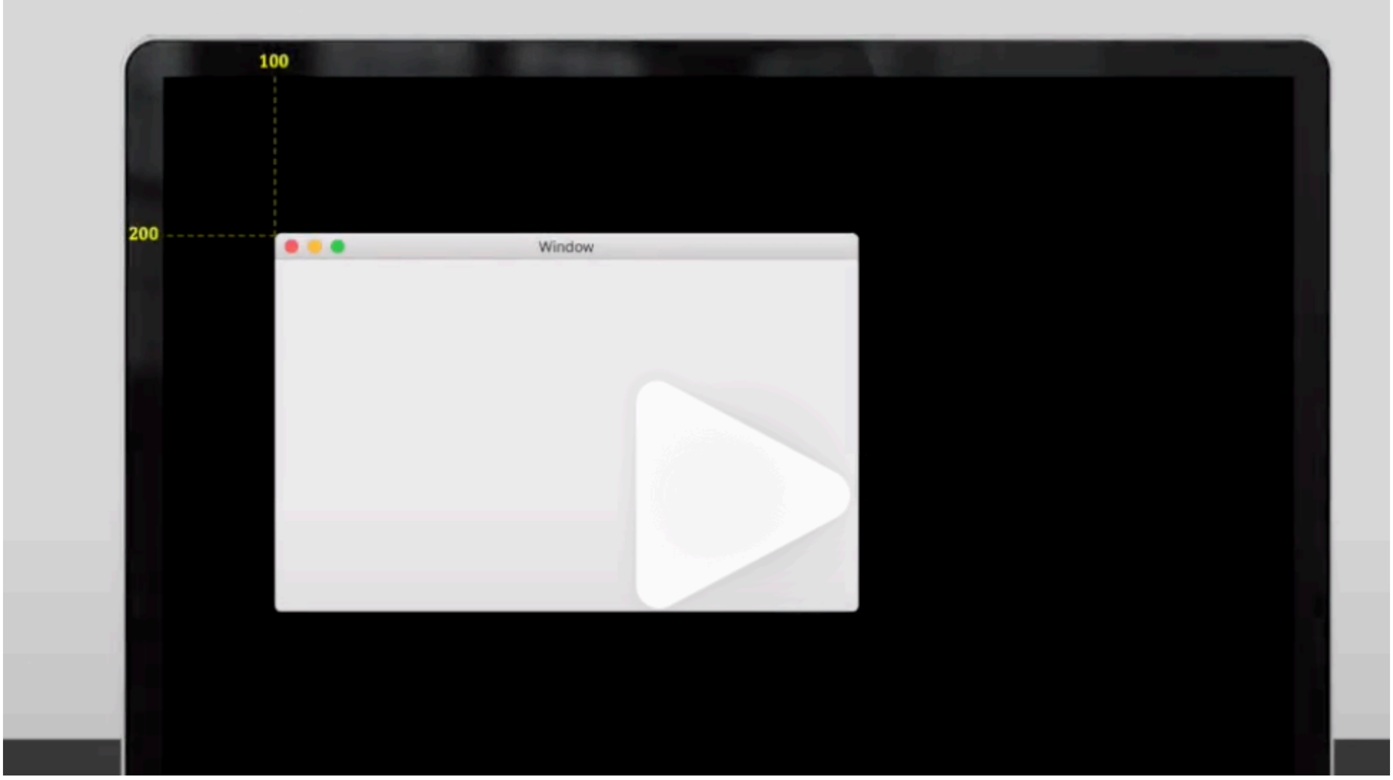
1. Initialize

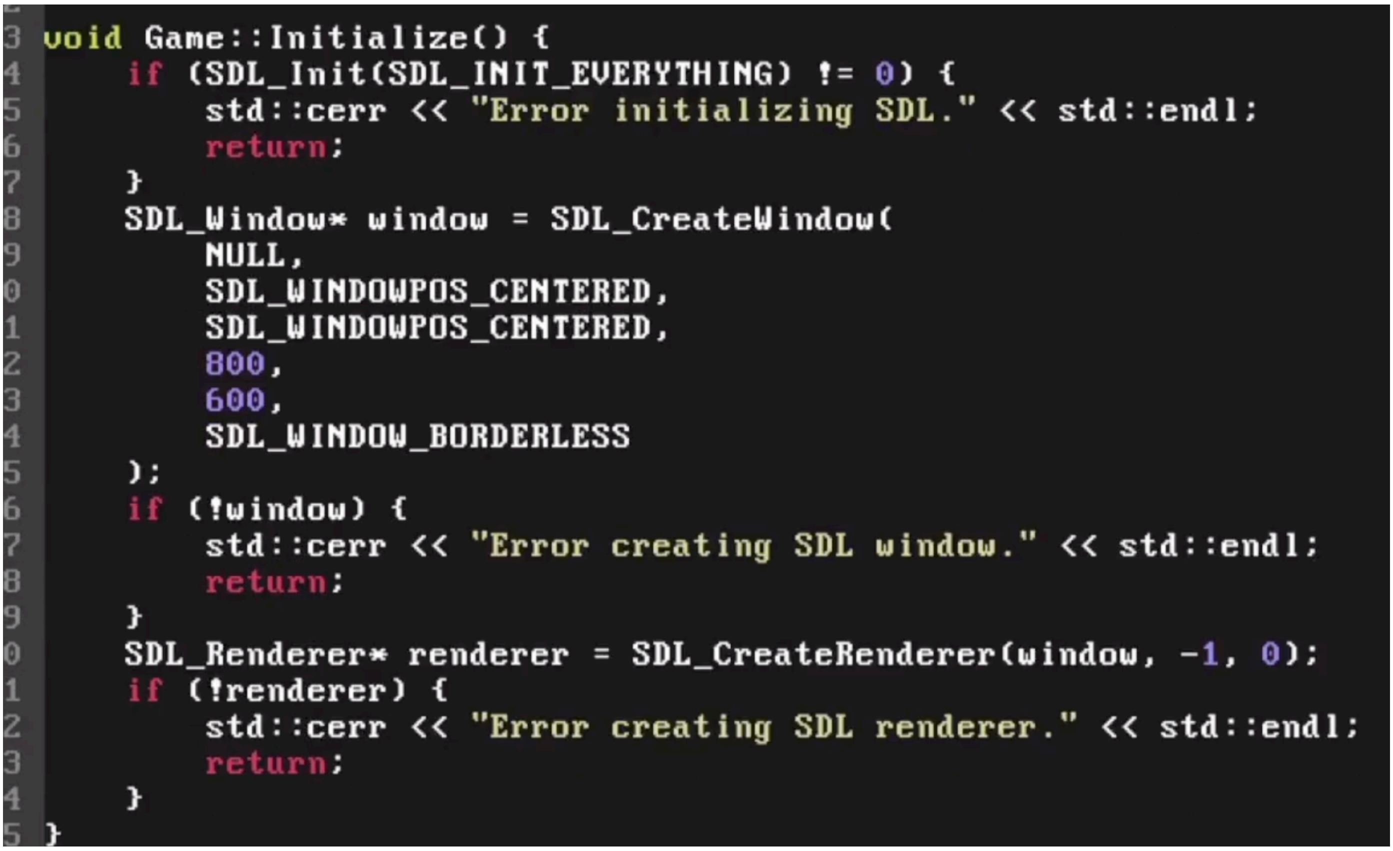
```
SDL_Init(uint constant);
// return 0: 正常
```

2. Functions of SDL

```
SDLSDL_CreateWindow(title,
                 posX(200), posY(100)hard code,
                 // position: use SDL_WINDOWPOS_CENTERED based on screen's resolution
                 800,
                 600,
                 SDL_WINDOW_BORDERLESS
                );
返回true则成功创建

```



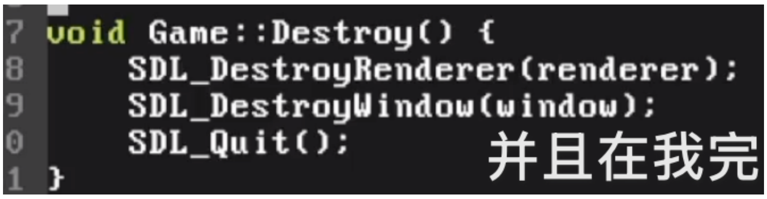


```
void Game::Initialize() {
    if (SDL_Init(SDL_INIT_EVERYTHING) != 0) {
        std::cerr << "Error initializing SDL." << std::endl;
        return;
    }
    SDL_Window* window = SDL_CreateWindow(
        NULL,
        SDL_WINDOWPOS_CENTERED,
        SDL_WINDOWPOS_CENTERED,
        800,
        600,
        SDL_WINDOW_BORDERLESS
    );
    if (!window) {
        std::cerr << "Error creating SDL window." << std::endl;
        return;
    }
    SDL_Renderer* renderer = SDL_CreateRenderer(window, -1, 0);
    if (!renderer) {
        std::cerr << "Error creating SDL renderer." << std::endl;
        return;
    }
}
```

3.Destroy

SDL_Window*\Renderer * local无法正常析构！

放到Game的全局private成员中，所有成员函数都能访问这些属性



```
void Game::Destroy() {
    SDL_DestroyRenderer(renderer);
    SDL_DestroyWindow(window);
    SDL_Quit();
}
```

4.Main：Run()

isRunning

停不下来：增加bool isRunning；

默认false，Initialize之后若都成功，isRunning设置为true；

```
while (isRunning) {
  ProcessInput();
  ...
}
```

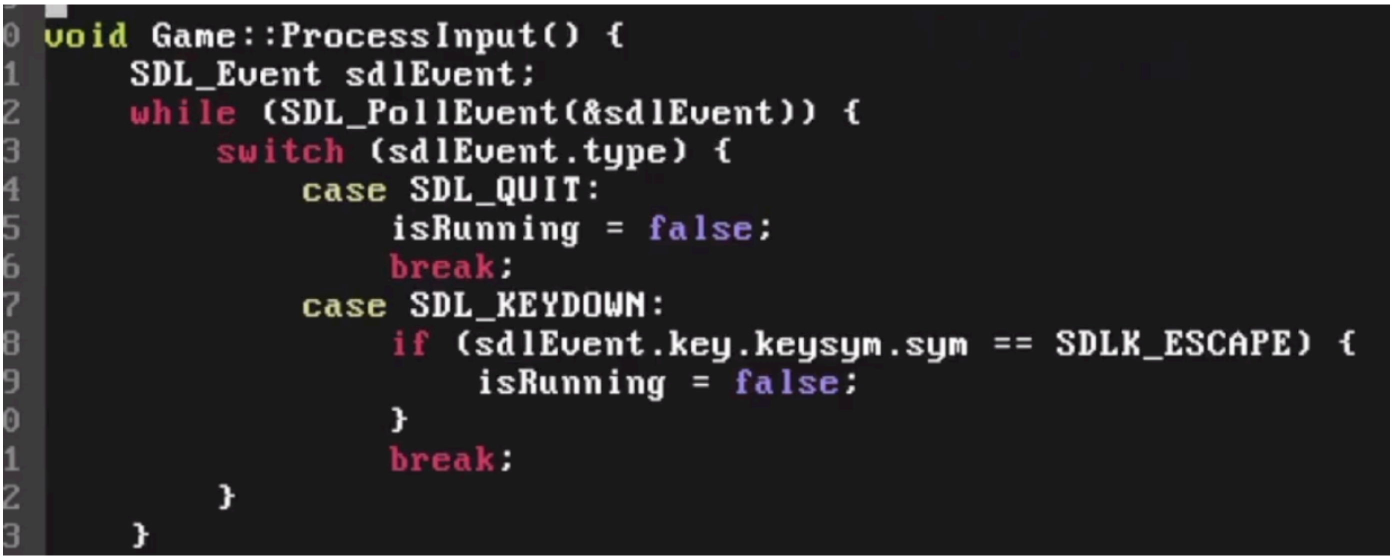
ProcessInput（）

SDL_Event;

一帧内可能有多个event：使用while处理

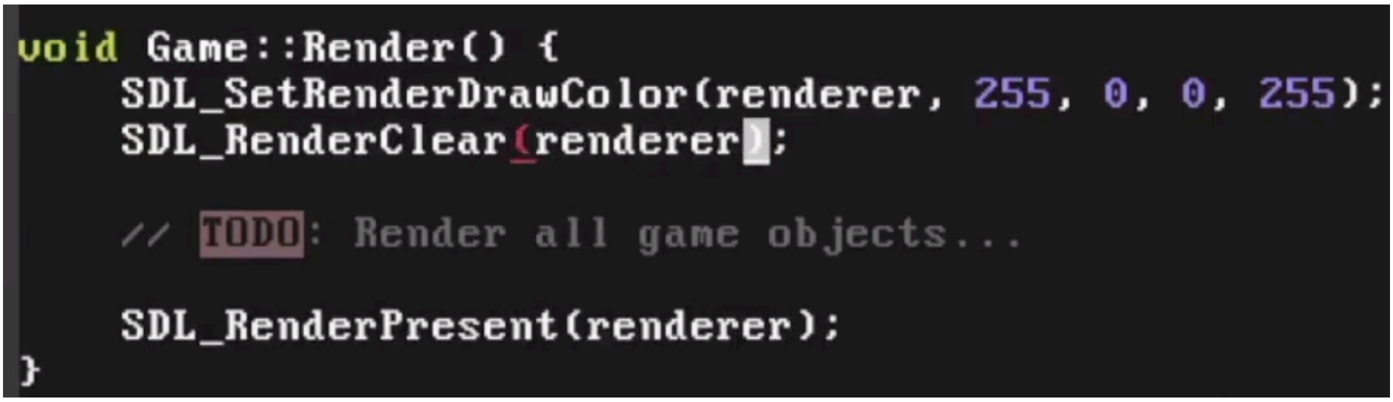
type：

SDL_QUIT: 用户尝试关闭窗口时系统发出事件



```
void Game::ProcessInput() {
    SDL_Event sdlEvent;
    while (SDL_PollEvent(&sdlEvent)) {
        switch (sdlEvent.type) {
            case SDL_QUIT:
                isRunning = false;
                break;
            case SDL_KEYDOWN:
                if (sdlEvent.key.keysym.sym == SDLK_ESCAPE) {
                    isRunning = false;
                }
                break;
        }
    }
```

5.Render()



```
void Game::Render() {
    SDL_SetRenderDrawColor(renderer, 255, 0, 0, 255);
    SDL_RenderClear(renderer);

    // TODO: Render all game objects...

    SDL_RenderPresent(renderer);
}
```