



链滴

7 章 01

作者: [heyang5188](#)

原文链接: <https://ld246.com/article/1502760977072>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

```

// C++primer.cpp : 定义控制台应用程序的入口点。
//
#include "stdafx.h"
#include <iostream>
#include <string>
#include <vector>
using namespace std;
class Screen;
class Window_mgr
{
public:

    void clear();

};
class Screen {
    friend void Window_mgr::clear();
    unsigned height = 0, width = 0;
    unsigned cursor = 0;
    string contents;
public:
    Screen() = default;
    Screen(unsigned ht, unsigned wd) :height(ht), width(wd), contents(ht*wd, ' ') {}
    Screen(unsigned ht, unsigned wd, char c) :height(ht), width(wd), contents(ht*wd, c) {}

    Screen& move(unsigned r, unsigned c)
    {
        cursor = r*width + c;
        return *this;
    }
    Screen& set(char ch)
    {
        contents[cursor] = ch;
        return *this;
    }
    Screen& set(unsigned r, unsigned c, char ch)
    {
        contents[r*width + c] = ch;
        return *this;
    }
    Screen& display()
    {
        cout << contents;
        return *this;
    }
};

void Window_mgr::clear()
{
    Screen myScreen(10, 20, 'X');
    cout << "Screen before clear:" << endl;
    cout << myScreen.contents << endl;
    myScreen.contents = " ";
    cout << "Screen after clear:" << endl;
}

```

```

    cout << myScreen.contents << endl;
}
int main()
{
    Window_mgr w;
    w.clear();
    system("pause");
    return 0;
}
#include "stdafx.h"
#include <iostream>
#include <fstream>
#include <string>
#include <vector>
using namespace std;
istream& istext(istream& in)
{
    int v;
    while (in >> v, !in.eof()) //直到遇到文件结束符号才停止读取
    {
        if (in.bad())
            throw runtime_error("IO流错误");
        if (in.fail())
        {
            cerr << "数据错误, 请重试: " << endl;
            in.clear();
            in.ignore(100, '\n');
            continue;
        }
        cout << v << endl;
    }

    in.clear();
    return in;
}
int main()
{
    string x;
    vector<string> vecStr;
    ifstream in("shader.vs");
    while (in >> x)
    {
        vecStr.push_back(x);
    }
    for (auto &vec : vecStr)
        cout << vec << endl;
    system("pause");
    return 0;
}

```