

建议的开始流程

0. 先成功编译一次

在你开始写第一行代码之前，试着先成功编译并打开我们提供的代码框架。你应当能看到游戏的标题画面。在按回车开始游戏之后，你会看到黑屏的游戏画面。此时你看到的就是正在运行中的游戏，只是游戏里面什么也没有。

1. 背景

你要做的第一件事便是显示出“游戏画面”，也就是背景。你需要为背景创建你的第一个类，它将继承 `GameObject`。在这时，你大可不必立刻规划好“什么成员属于 `GameObject` 基类，什么成员应当单独存储”。在你实现更多功能的过程中，这个问题会渐渐变得清晰。

创建完背景类后，你需要在 `GameWorld` 中生成一个背景。在 `Init()` 中生成一个临时变量当然是不行的，即使这个临时变量是给它动态分配内存的 `shared_ptr`，也会因为离开 `Init()` 作用域（scope）时被销毁，然后分配的动态内存也会释放。

你应当把这个背景存储在 `GameWorld` 里。根据文档，所有 `GameObject` 都应存储在一个 `list` 里。考虑一下这个 `list` 的类型会是什么？

一切顺利的话，现在运行游戏你就能看到背景图片。如果你遇到编译错误或链接（linking）错误，看看在用到背景类的时候是否正确 `#include` 了它的文件？参考 FAQ 中的 Linking error 部分。

2. 种一棵植物

接下来，你可以真正开始“玩游戏”了。如果你感觉无从下手，就从种植一棵植物（以向日葵为例）开始，相信你在实现完之后就能明白游戏中任何部分的开发流程。

完整的“种植物”步骤是：点击种子包，再点击种植位，就在该位置上生成一棵向日葵。但现在，可以先简化一下：不要种子包了，点击地上的种植位就能种出植物。

那么，你首先需要能显示出来的向日葵（尽管它不能生产阳光），还需要可以点击的种植位。我们还是推荐你像背景一样，为他们分别创建代码文件或 CMake 目标，不要忘了解决链接问题。

种植位一样需要在 `GameWorld` 中生成，但他们是隐形的，没有贴图。为了方便 debug，你可以给它指定一个任意贴图。

现在就到了最核心的部分：如何在种植位的 `OnClick()` 里种出向日葵？直接新建的话，也会是一个会立刻销毁的临时变量。并且，只有存在于 `GameWorld` 的容器 `list` 中的物体，才会被显示/ `Update`。

种植位有权利管理植物吗？并没有。只有 `GameWorld` 才有权利。因此，是不是应该让种植位去“通知”它所在的 `GameWorld`，让 `GameWorld` 在指定位置上新建一个植物？你的种植位怎样才能“通知”到 `GameWorld`？（参考FAQ：`shared_from_this`）

3. 产生一些阳光

如果你成功地完成了上面的步骤，你应该对“`GameObject` 需要存储什么成员”有了更多的想法。试试继续让这个游戏动起来：增加天上掉落的阳光和向日葵产生的阳光。

在这时，你可以思考一些设计上的问题了。两种阳光有什么相同或不同？是分成两类去写还是做特殊判断？

4. 自由发挥

剩下的部分里，我相信你们已经不会遇到太多问题了，开始自由发挥吧！