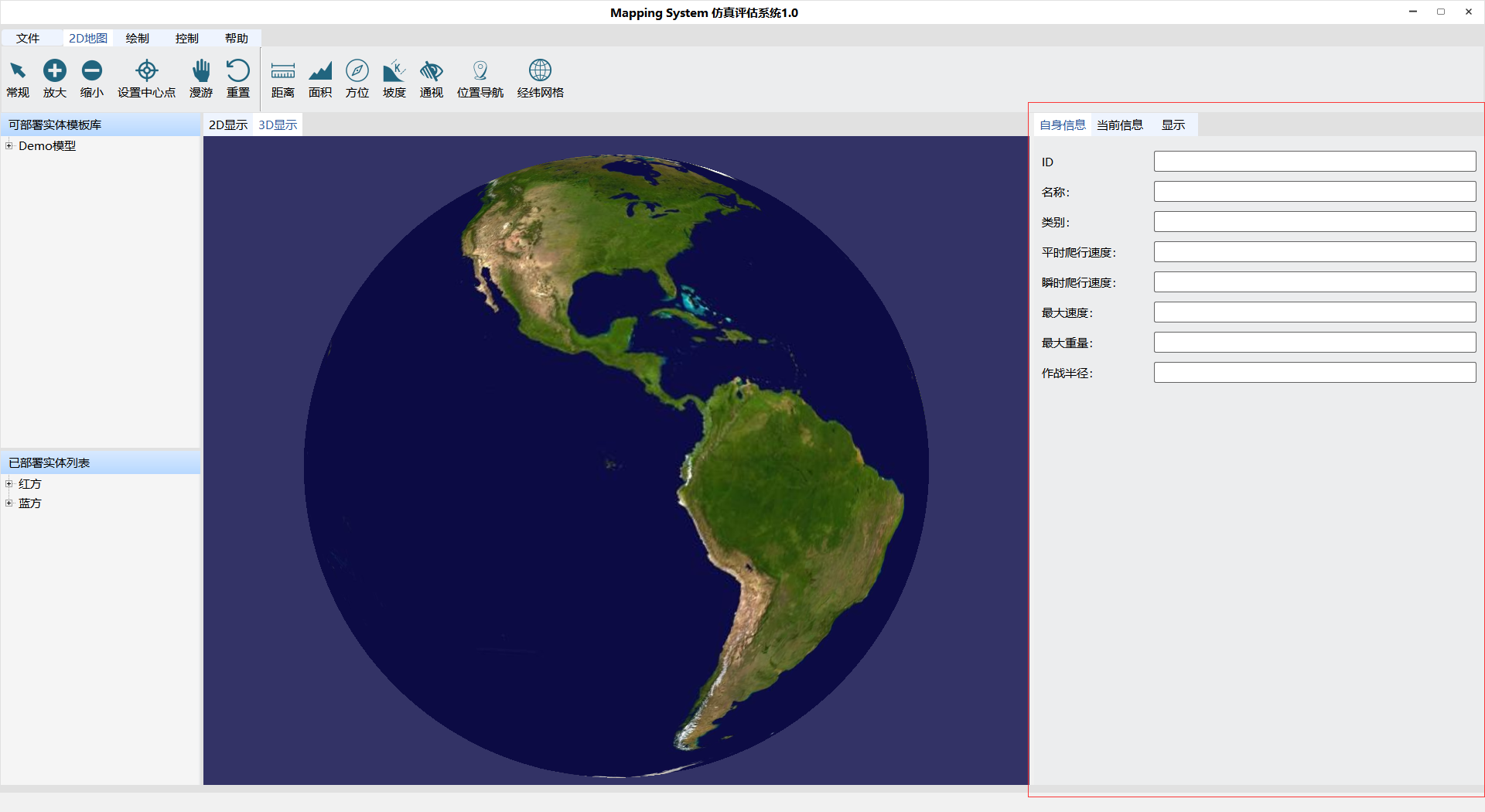
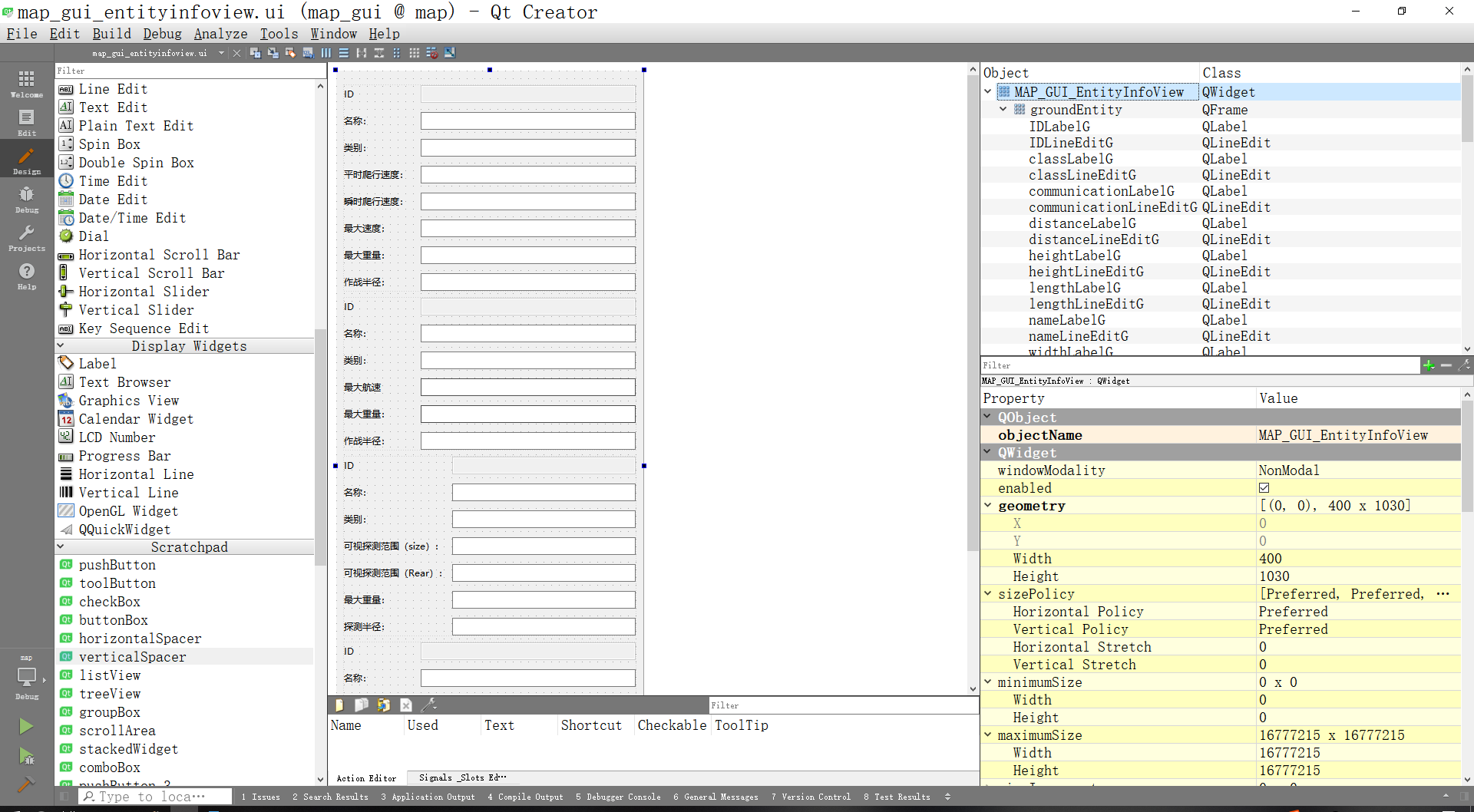
基于上一份界面设计开发框架，我们得到一个界面后，考虑它上面的功能以及与系统其它部分交互时应遵守一些原则，总体思路是依据MVC方法，实现解耦，降低复杂度，提高复用性。

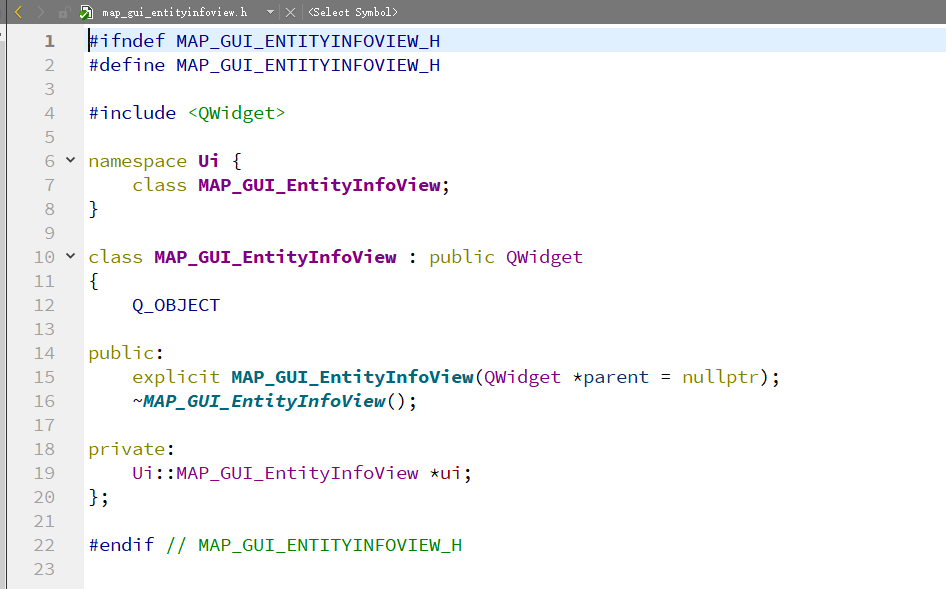
下面以一个子界面为例：



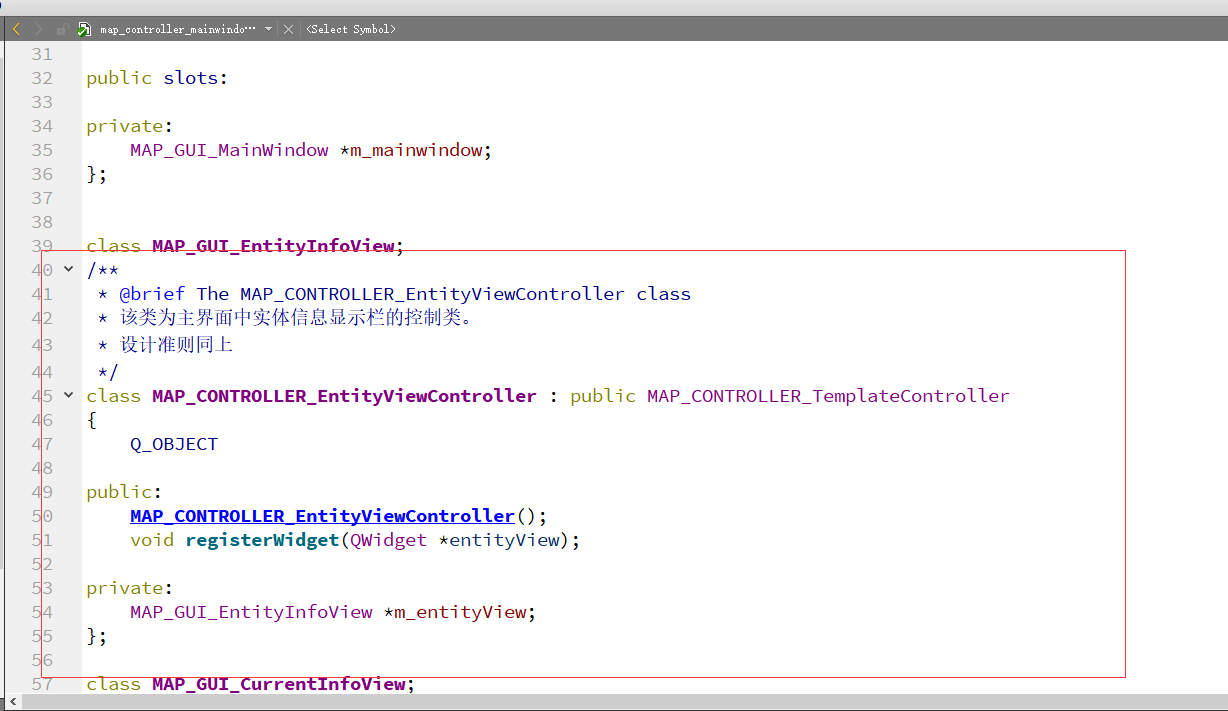
主界面右侧栏其中的一个tab，是单独的一个子界面

1. 生成Qt的ui/h文件，依据《界面设计》文档完成界面开发

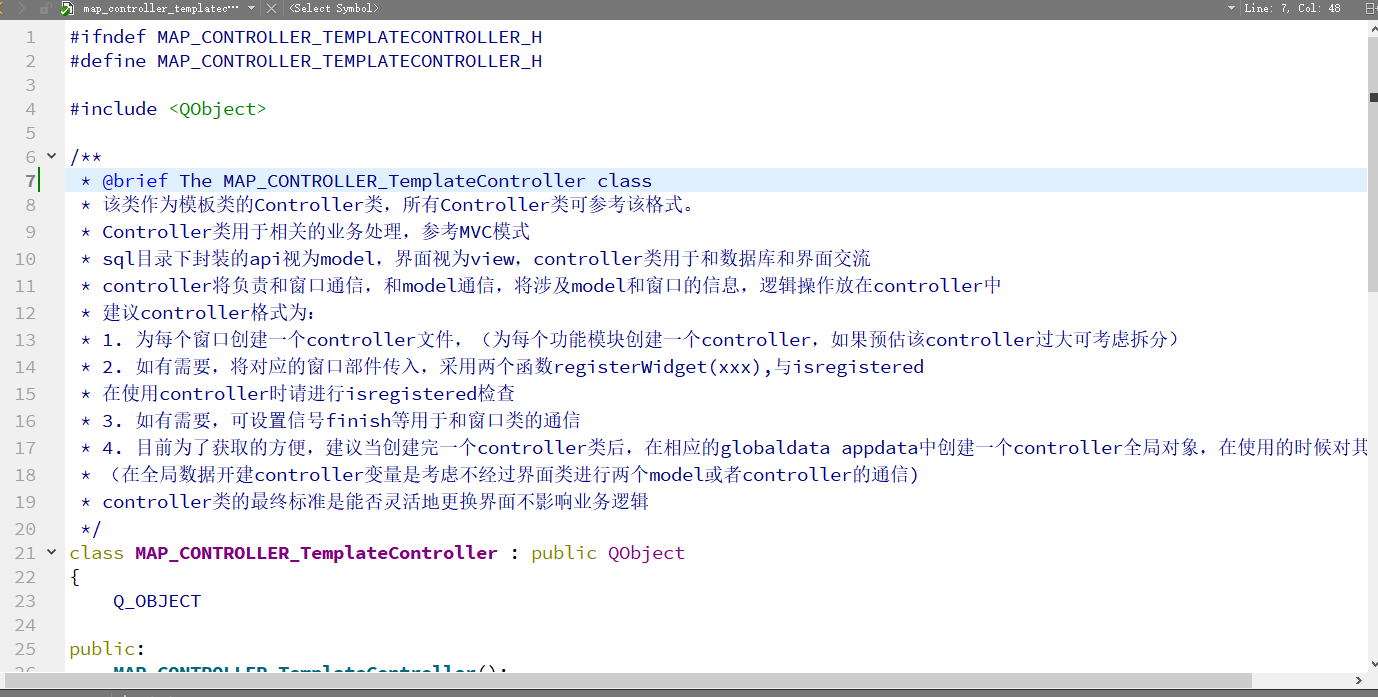




1. 为该界面添加一个控制器：



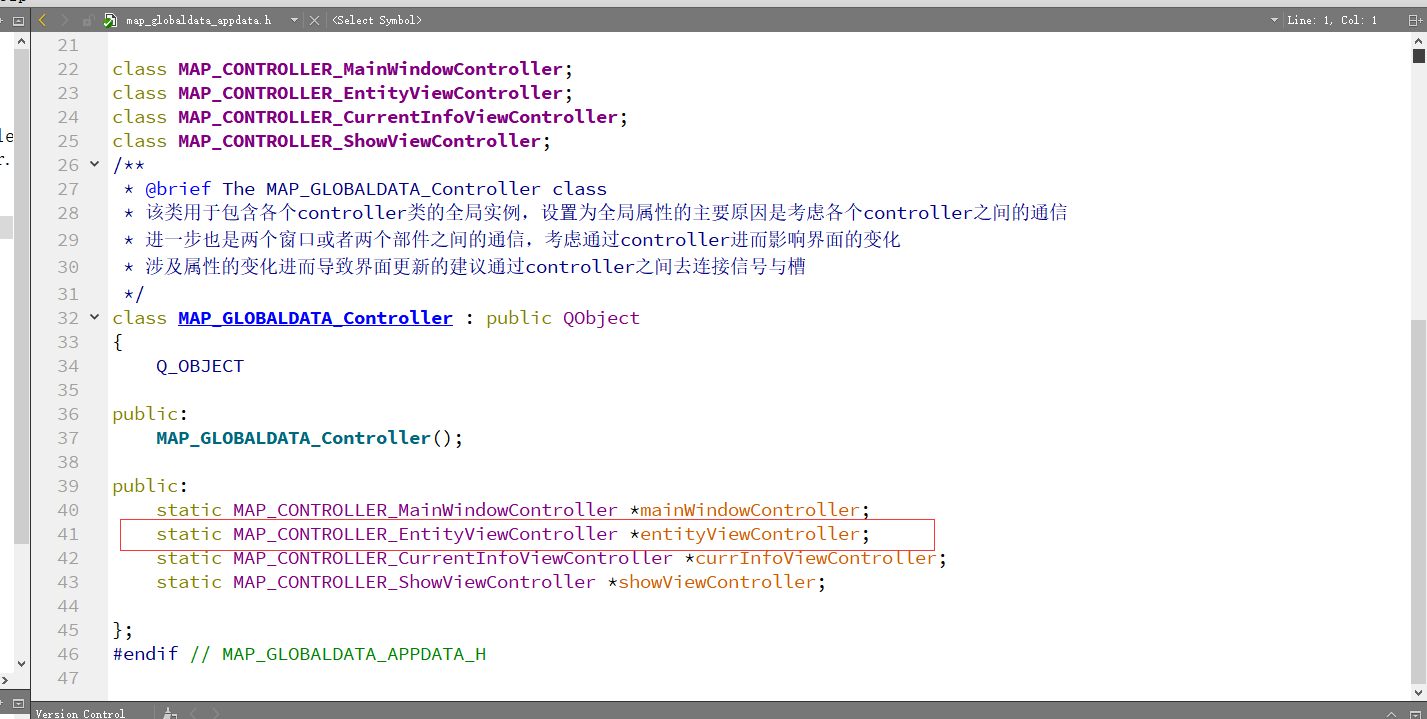
添加方法以及原则参考map\_controller/map\_controller\_templatecontroller.h，该类可作为基类使用。

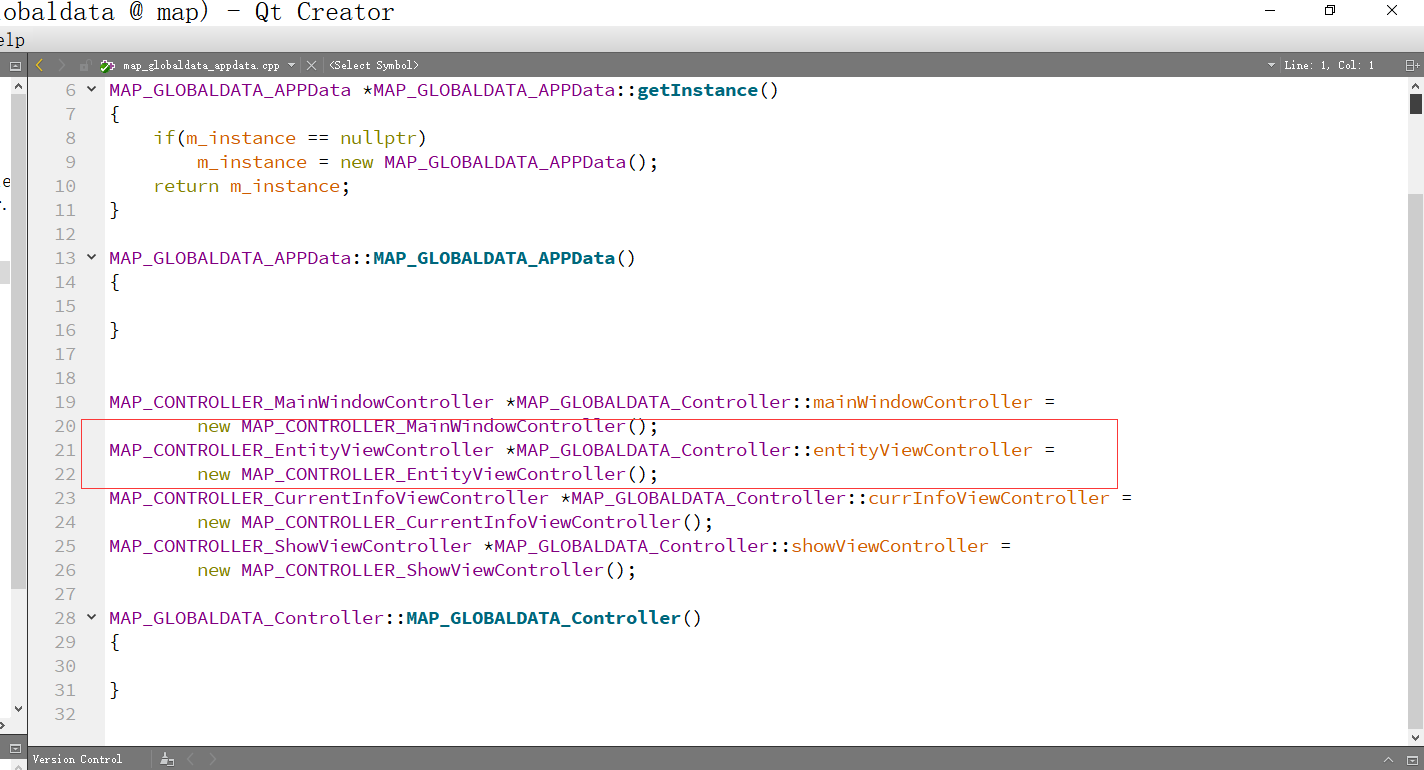


其中，添加控制器时如果该界面作为大界面的一个子部分，建议可以放在大界面的同一个控制器文件中（为避免控制器文件过多）。

详情请查看头文件的介绍。

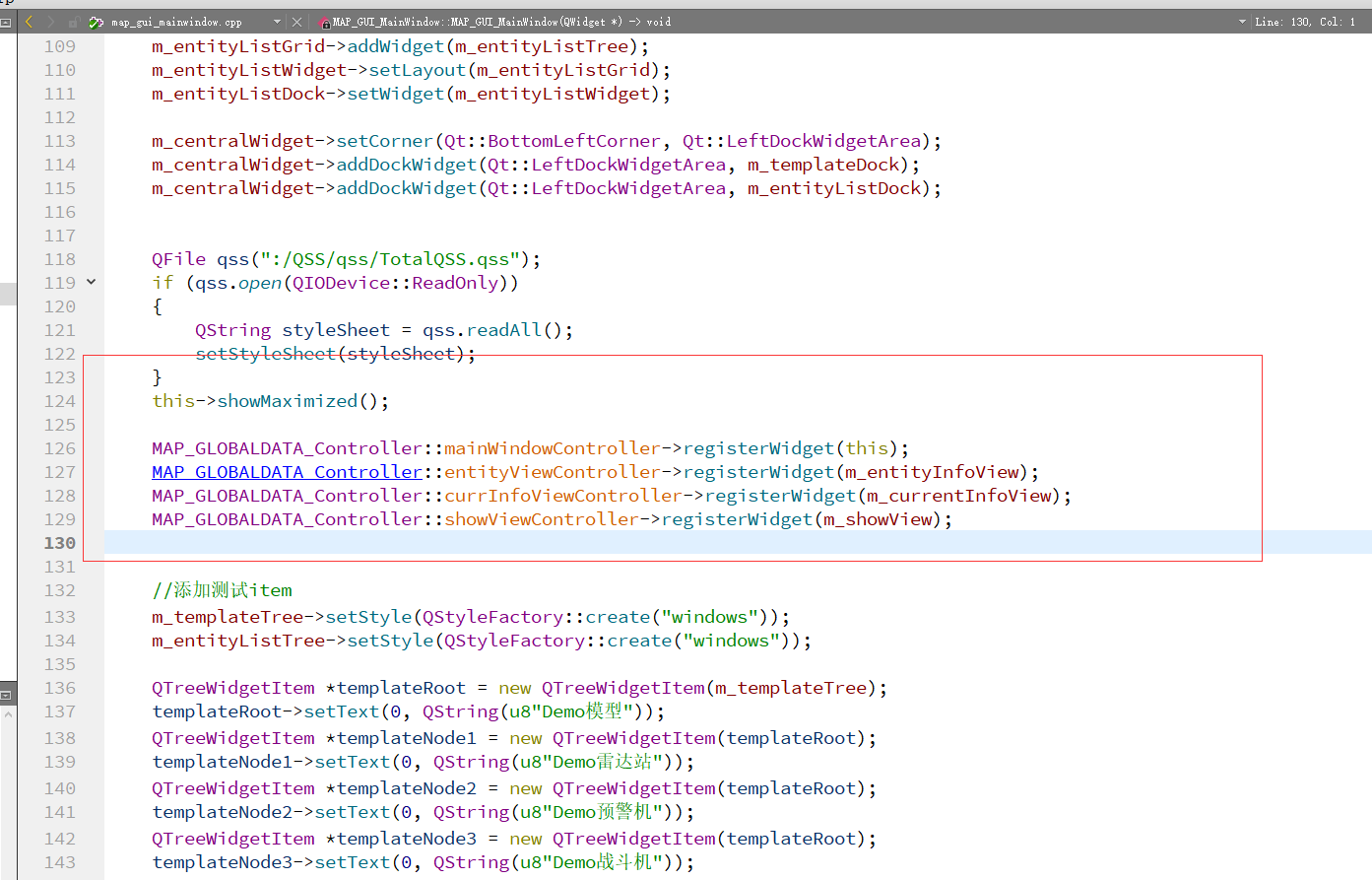
1. 在全局数据中添加并初始化该控制器：



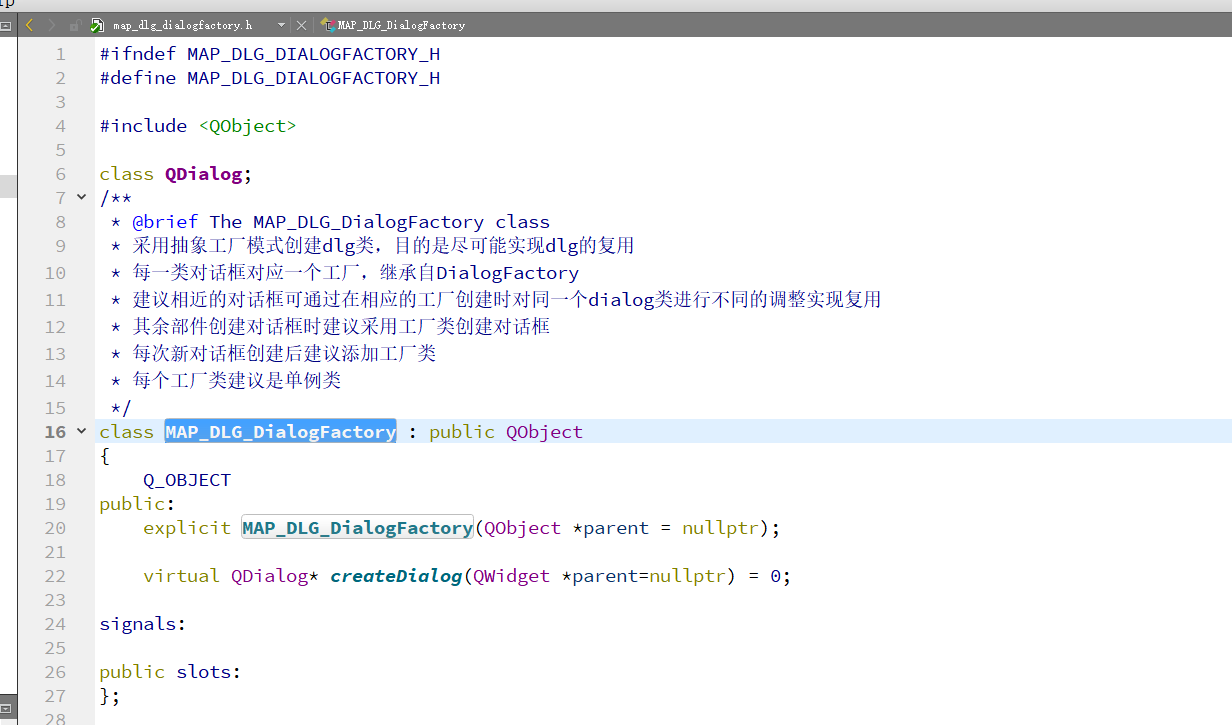


见于map\_globaldata/map\_globaldata\_appdata.h头文件。

1. 在创建该界面的地方注册该界面的控制器



1. 然后系统各部分之间的交互即可通过控制器进行，比如跨界面的信息传递等；同时避免在界面中出现过于复杂的业务逻辑，或者与其他多个类比如数据库等产生大量耦合。
2. 当然，在考虑复用、解耦等问题时，要平衡利弊，对于较为简单的界面，且功能集中单一，不存在复用的可能，也可以跳过控制器，直接通过QT本身的信号槽工作。
3. 所有对话框（即QDialog），通过工厂类的控制生成，见于： map\_dlg\map\_dlg\_dialogfactory.h：



实现复用，详情参考头文件的介绍。