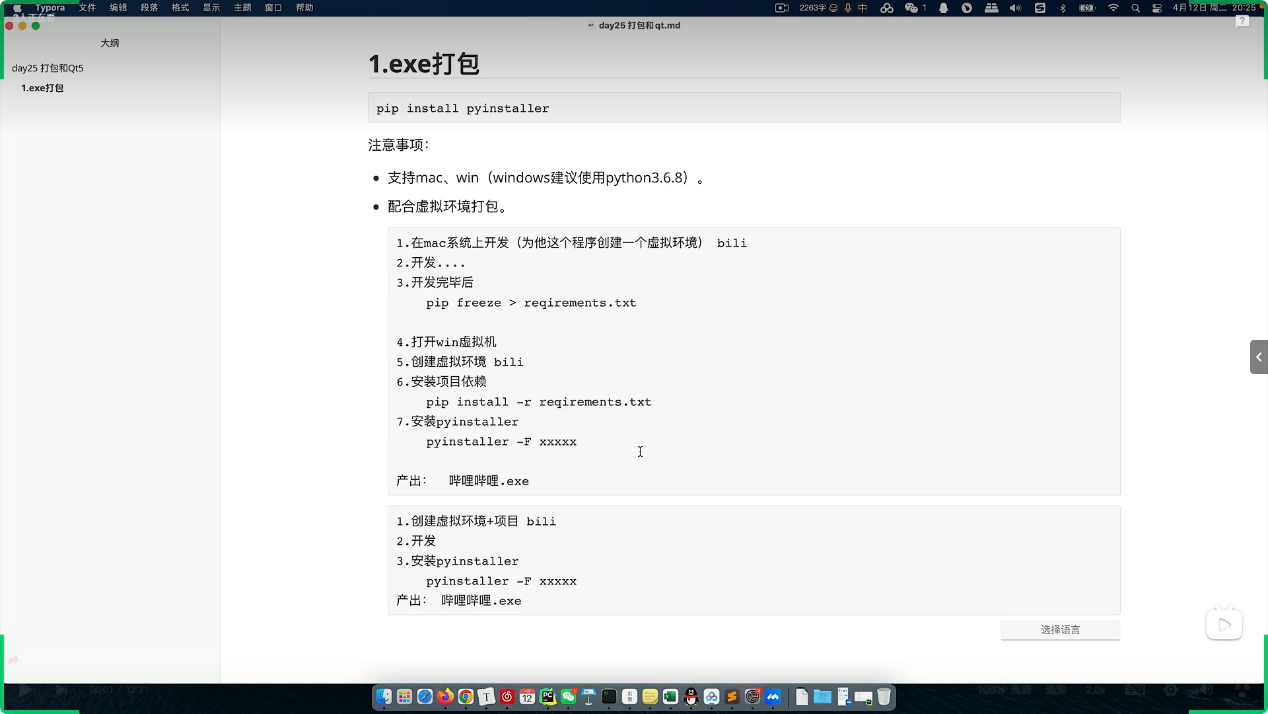
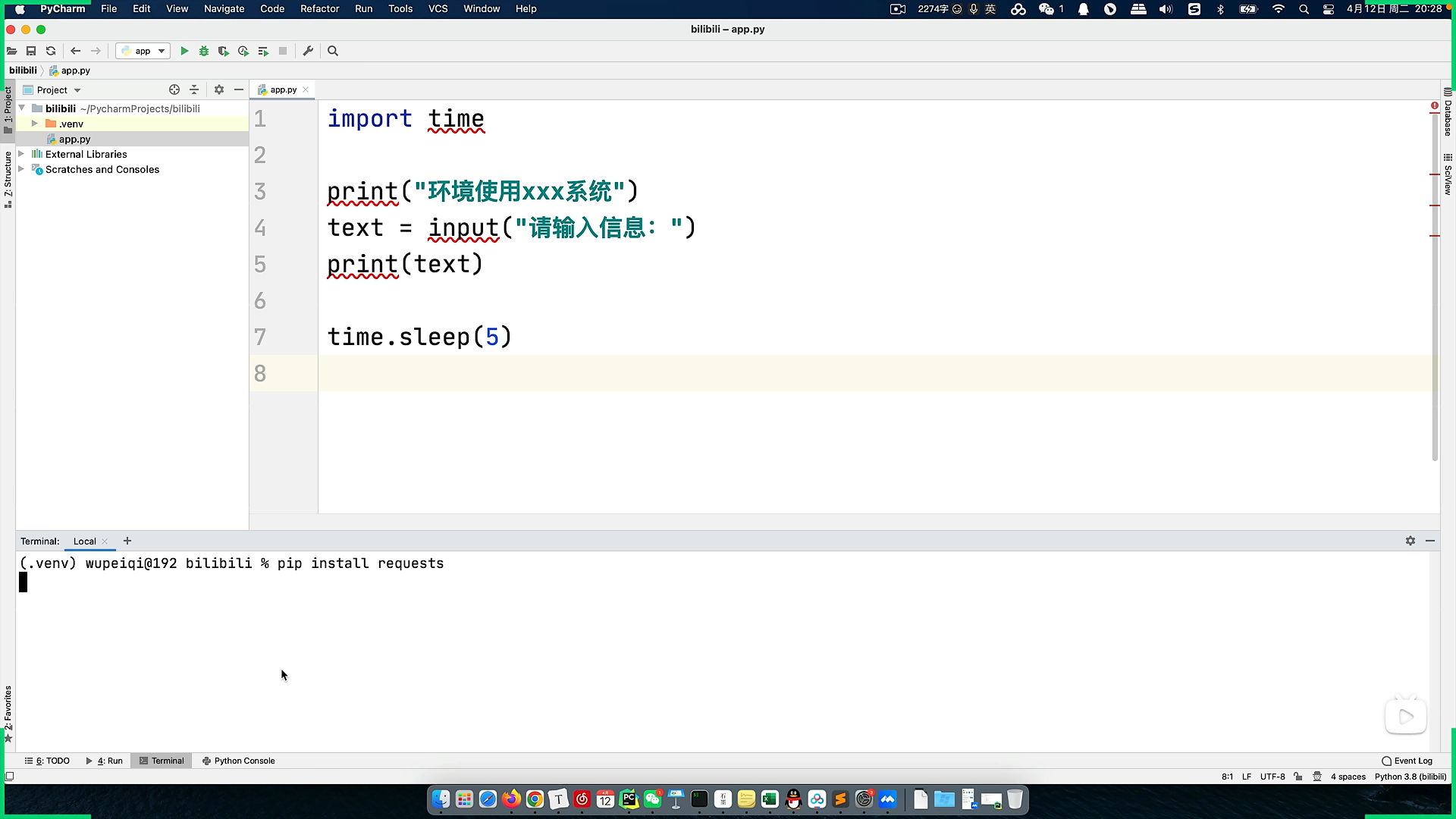
必须先安装pyinstaller这个包

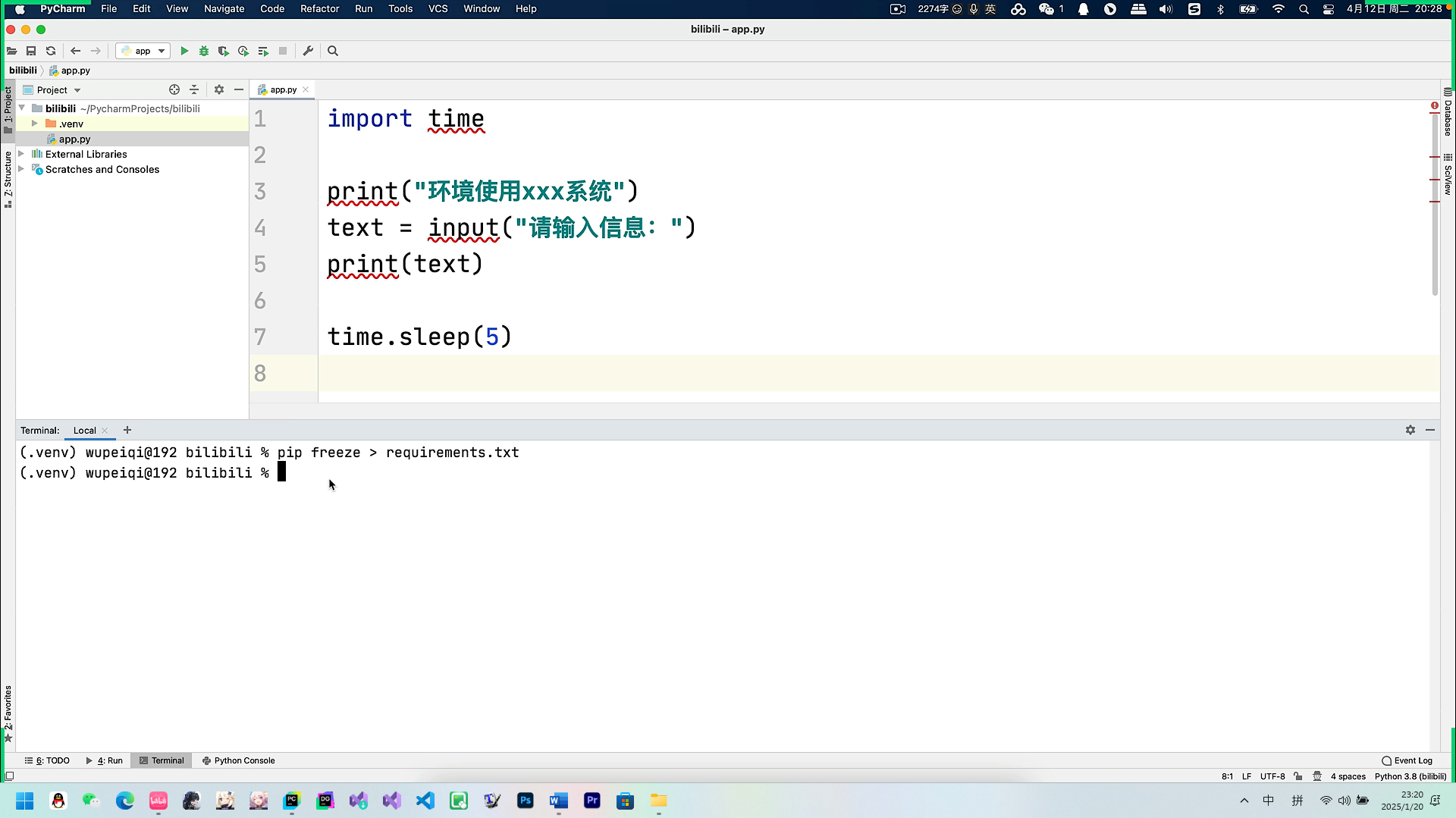
requirem 修正



安装依赖包



生成requirement.txt文件，文件里面写了需要的内部依赖



打包指令：pystaller -D 源代码文件全名

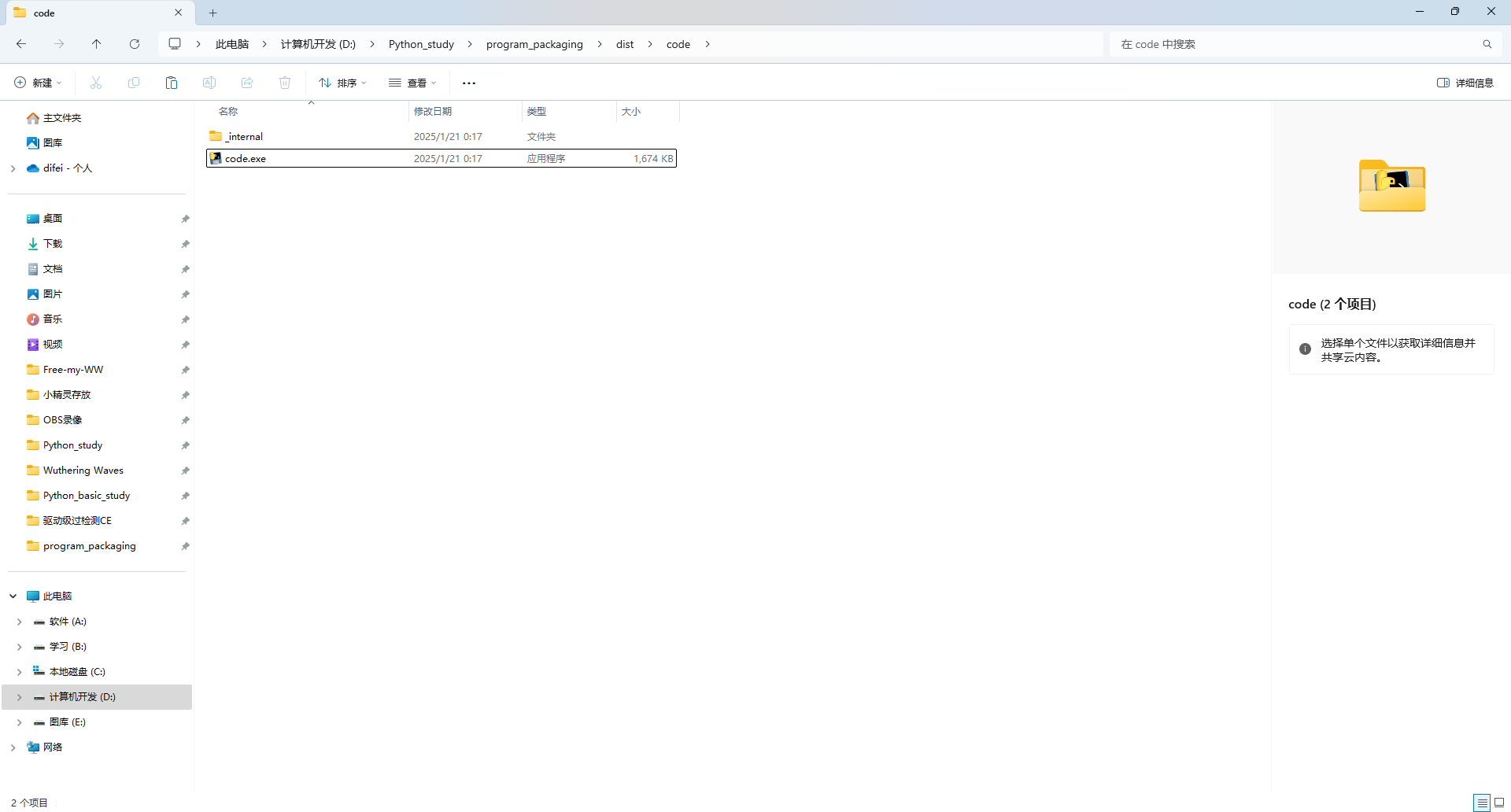


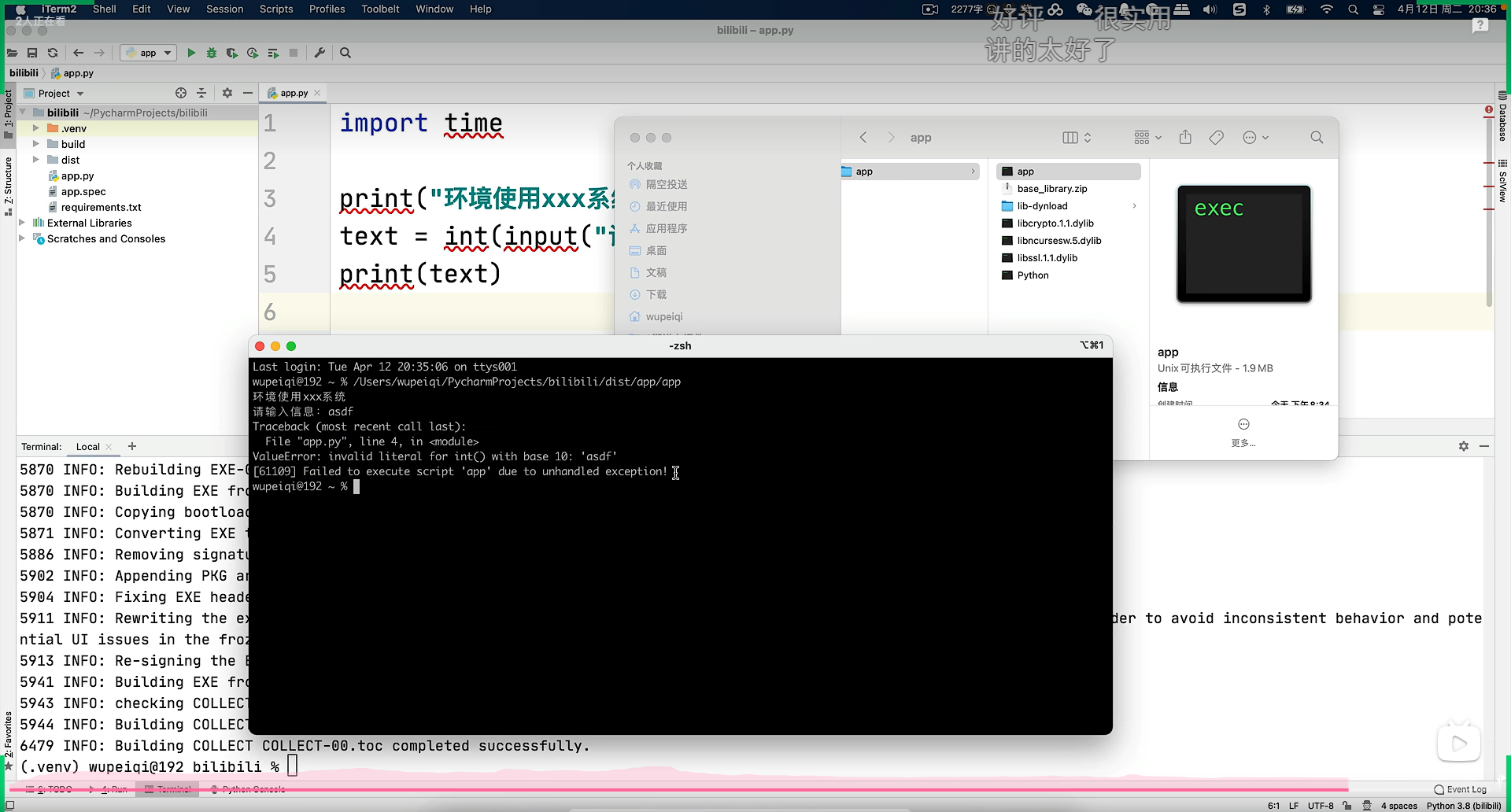
打包后会生成2个文件，1个文件夹：

build是打包编译过程产生文件，没有用

打包的源文件名.spec 这个文件是配置文件

dist 文件夹里面就是我们需要的文件了，里面有exe，点击就可以使用了，但是如果没有加入GUI的话就是黑框框了



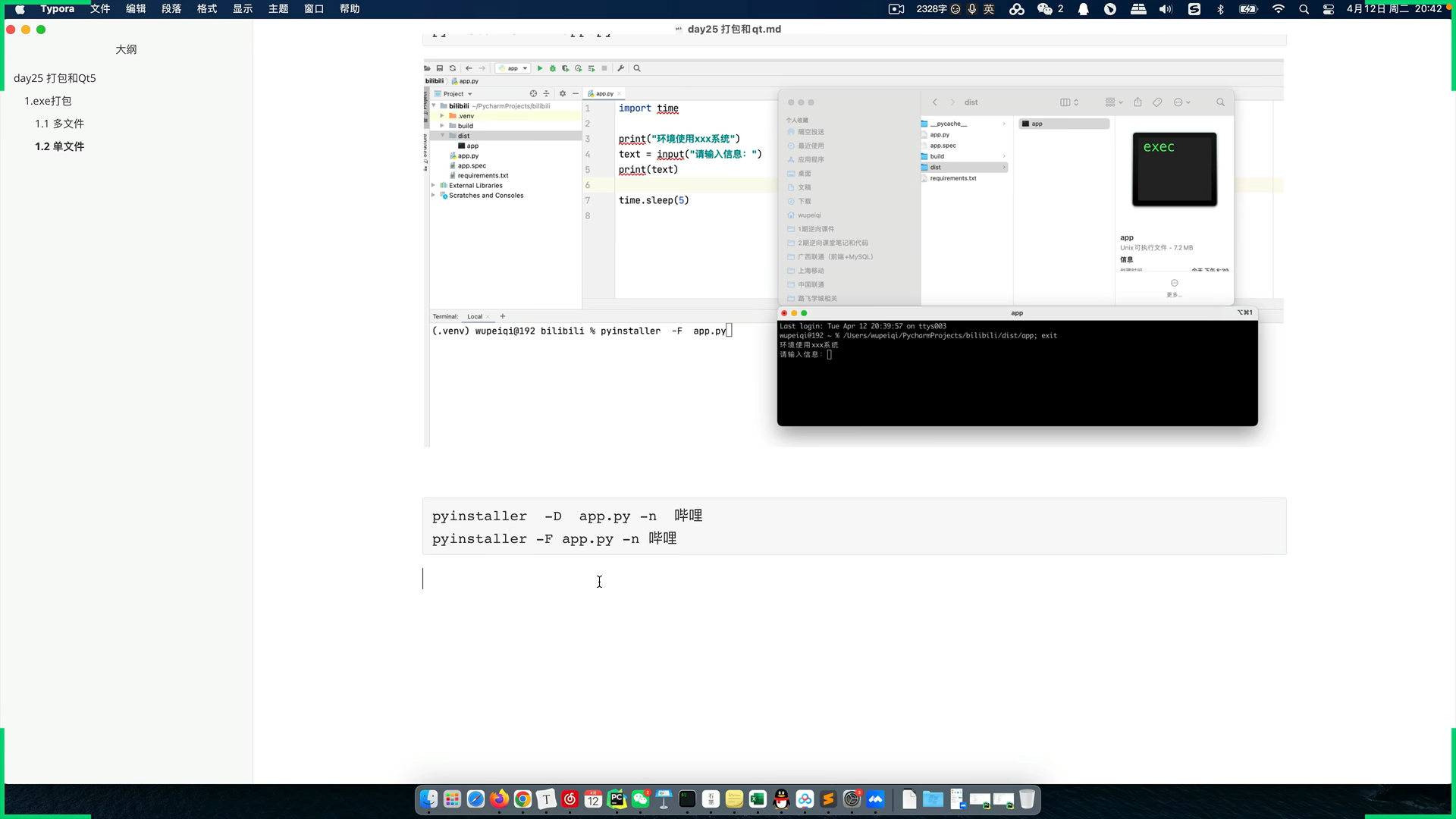
如果代码本身就有问题，运行到有问题的地方后就会直接关闭控制台（qt没测），可以放到cmd里面去跑代码，这样就能看到错误了。打开cmd后加路径就ok了

打包之后文件都放在一起，简称多文件，运行速度会快一些，以上就是多文件打包

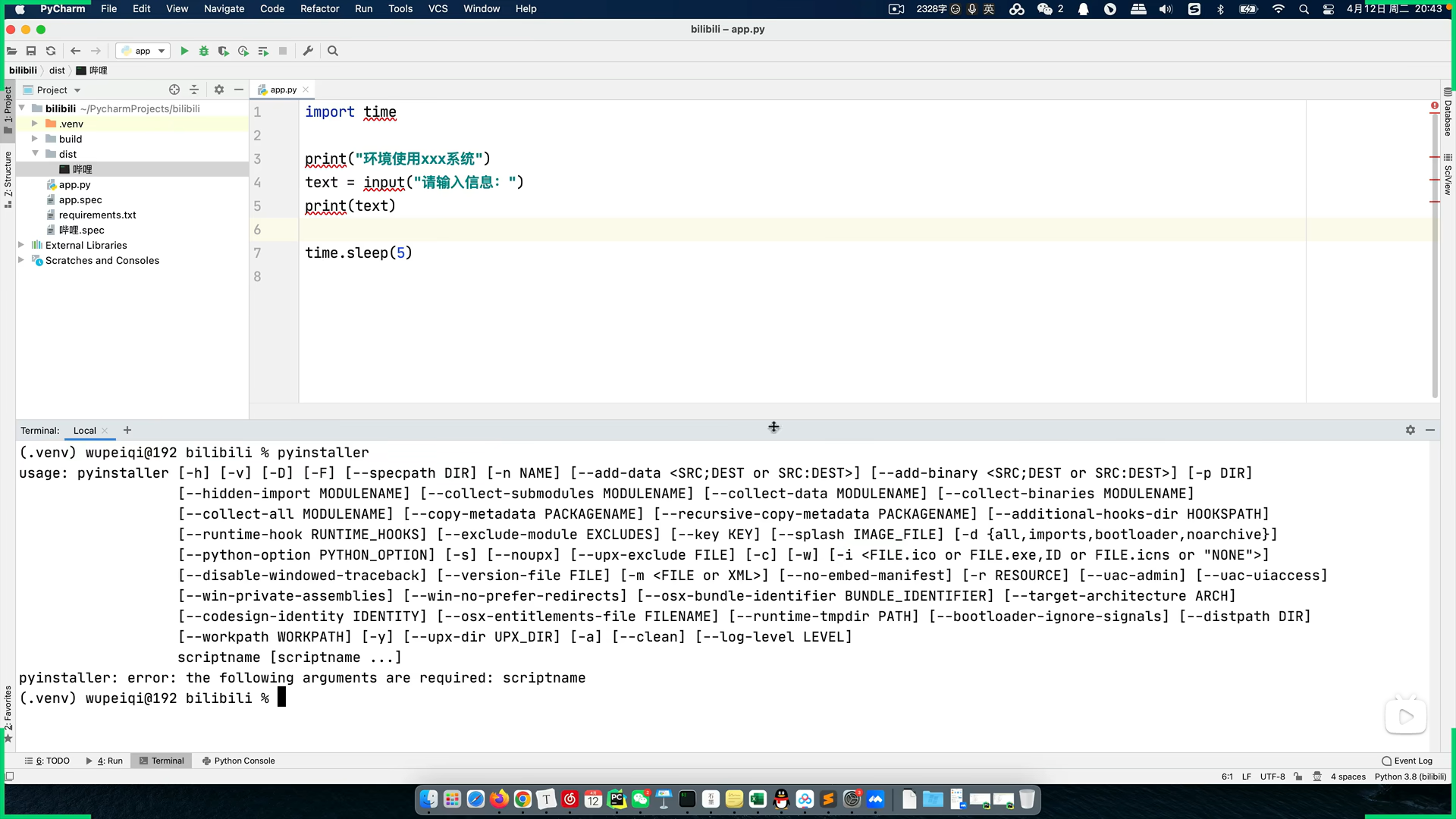
接下来记录的是单文件打包，考虑的是用户面对同一个目录下有很多个文件都不知道点哪一个文件启动。



在原来打包基础上加上“-n 修改的名字”这样就可以修改打包出来的名字了



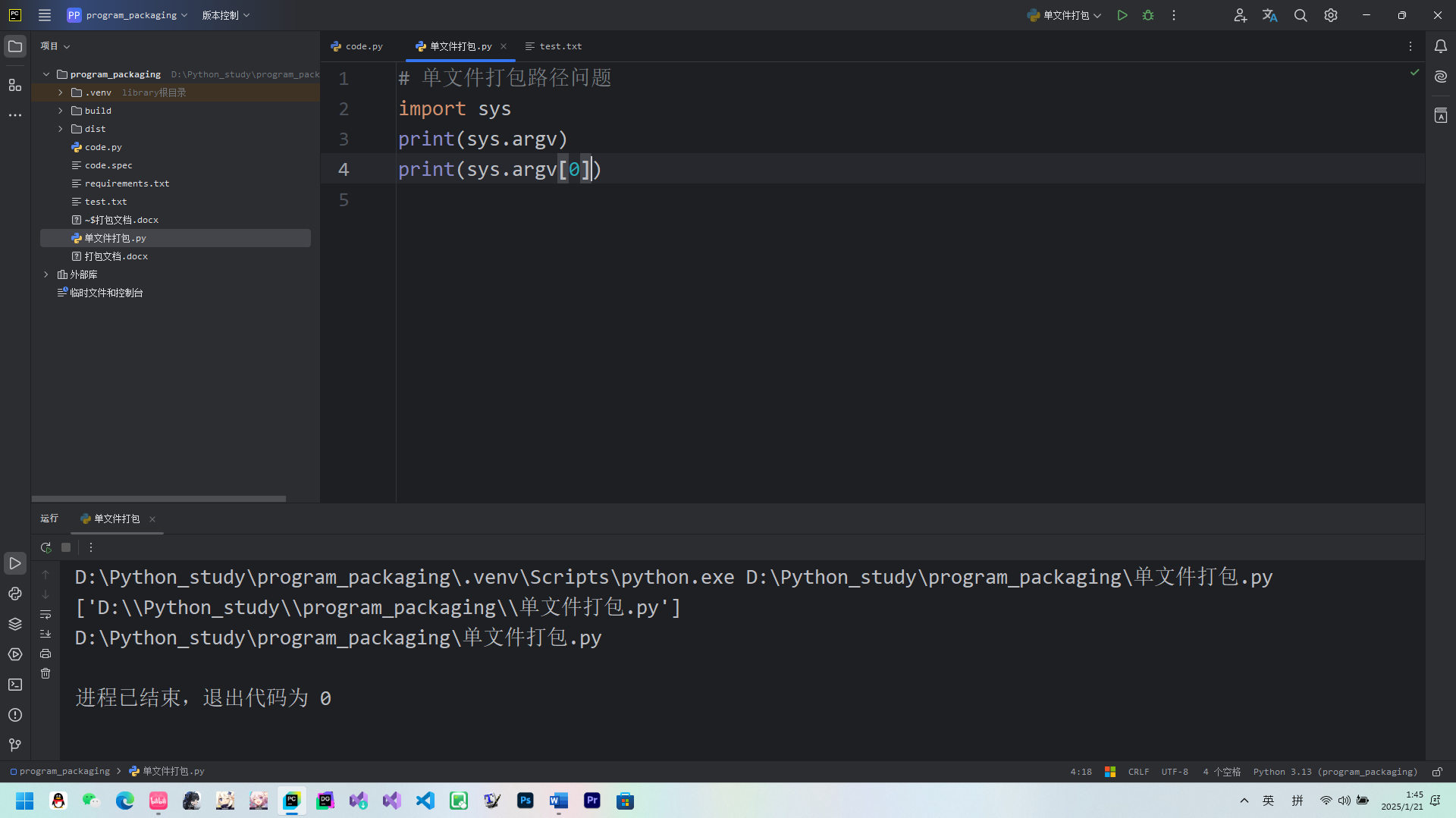
还有很多参数，比如可以修改打包出来的图片之类的的



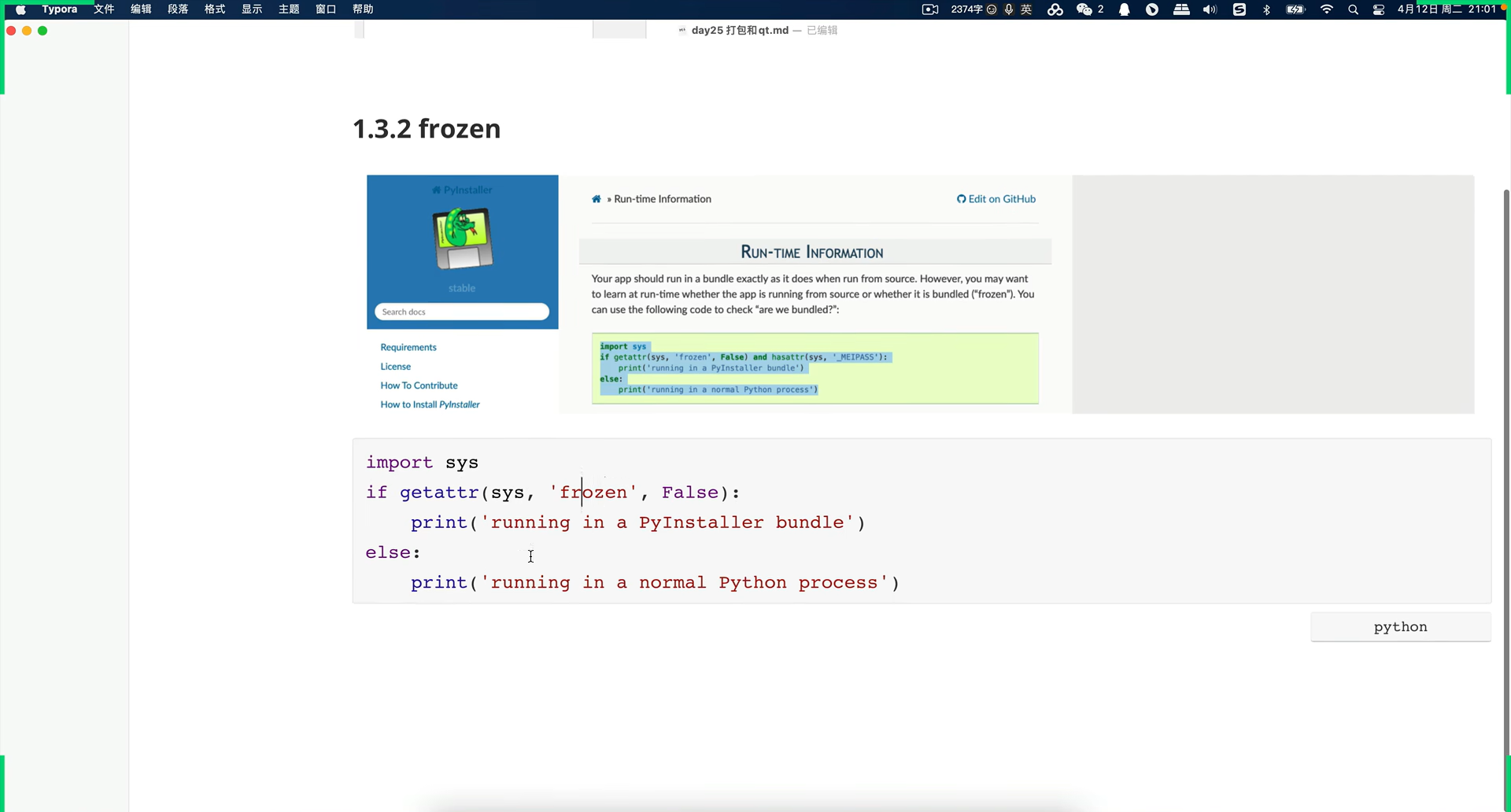
单文件打包会把需要的资源文件解压放到临时文件目录里面去（我觉得真傻逼）

如果是代码内置查找当前目录下的文件那就完蛋了，找不到临时目录下的文件。

如果还是要使用单文件打包的话就要改读取文件的目录了。调用sys的包来获取解压的目录()，这种方式代码内使用的路径和不同方式打包后的路径都稳



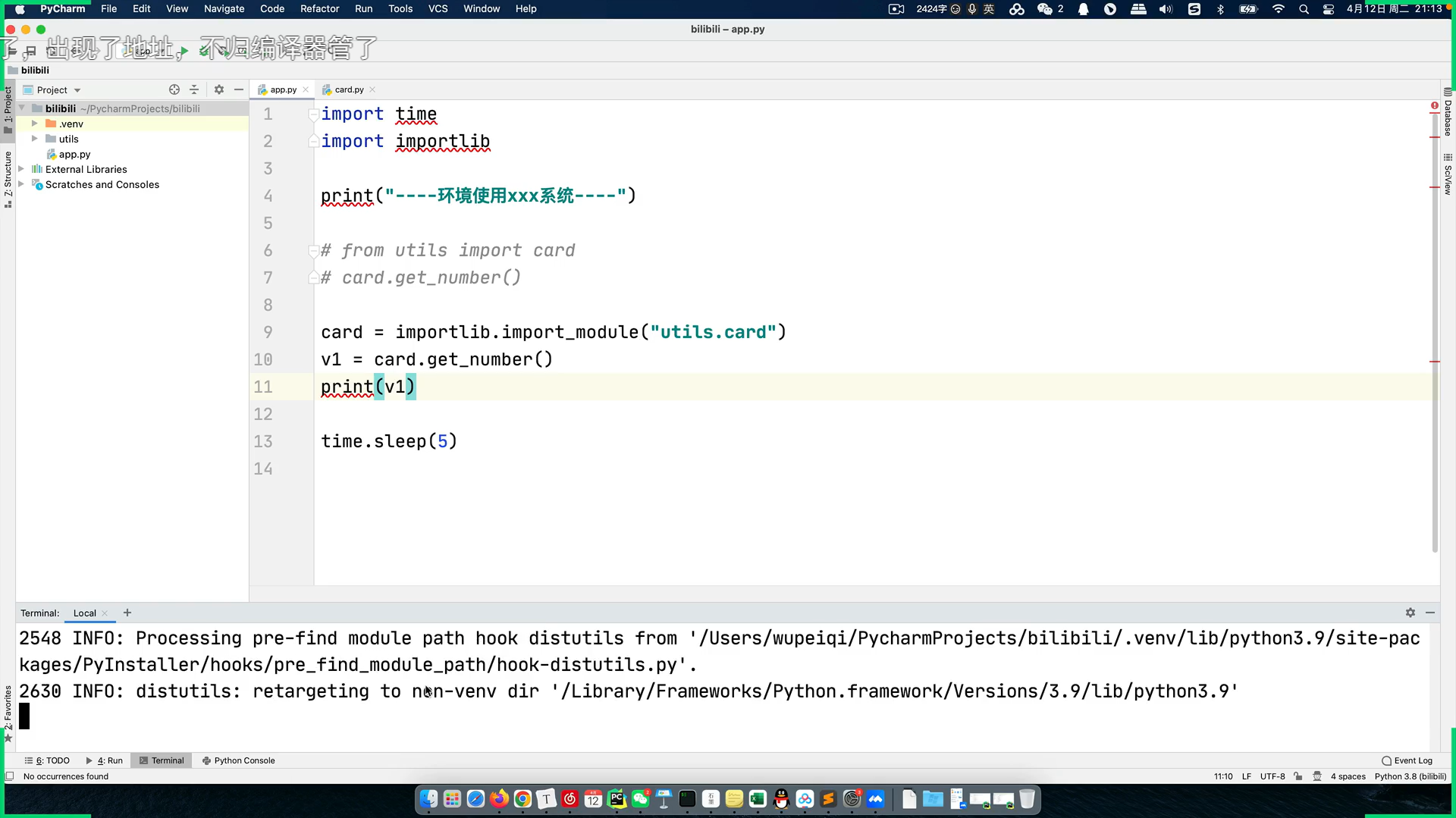
路劲问题解决方法2

使用不同的方式获取路径（上一个好）

模块问题

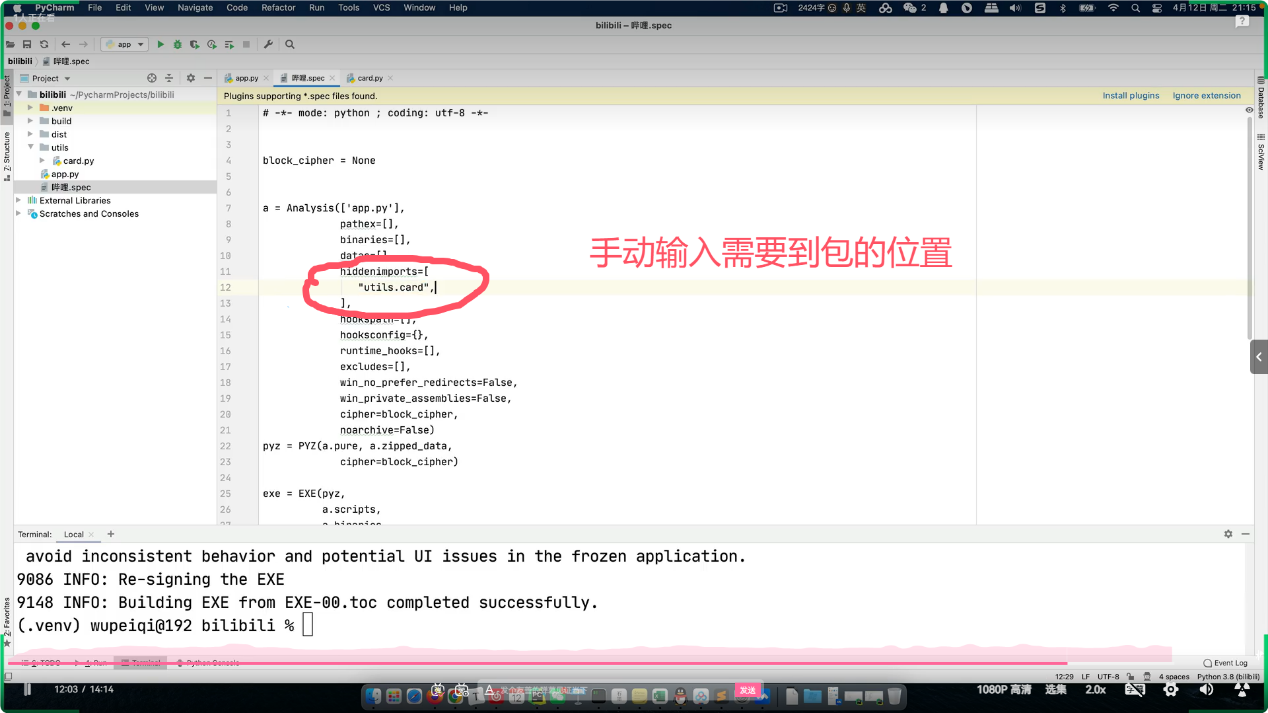
默认打包有相互依赖关系都会打包，不用担心自定义的包互相调用不打包的问题，如果打包文件里面存在没有用到的py包就不会打包。

还有就是如果遇到那种动态导入模块的代码时，他是无法找到关联的包。



遇到这种情况的时候就手动导包了

有个后缀是spec的文件，点进去，找到hiddenimport（如下图）输入手动导入的包名



收到导入之后一定要以这个方式打包，不然导包的配置会被覆盖掉的，千万不要拿py的文件打包了

