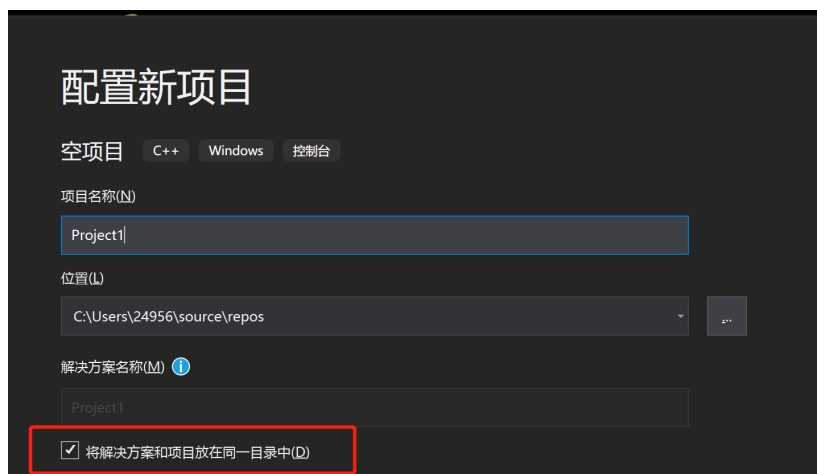
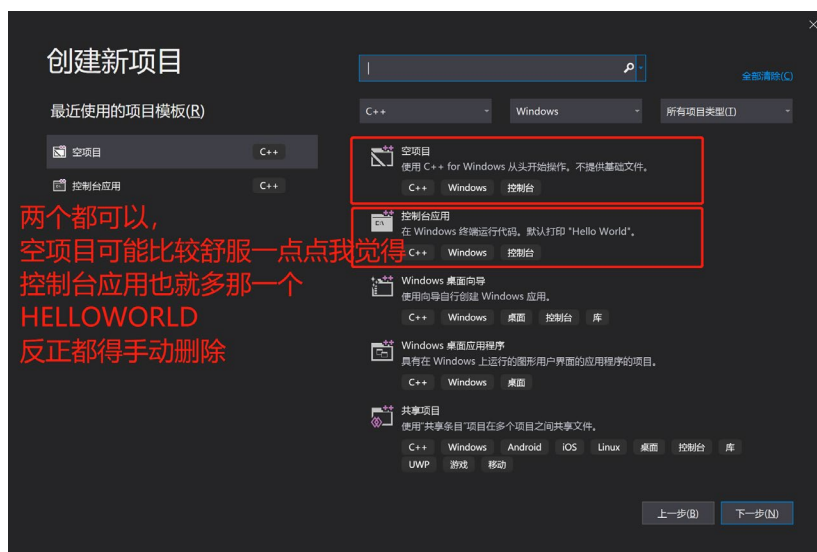
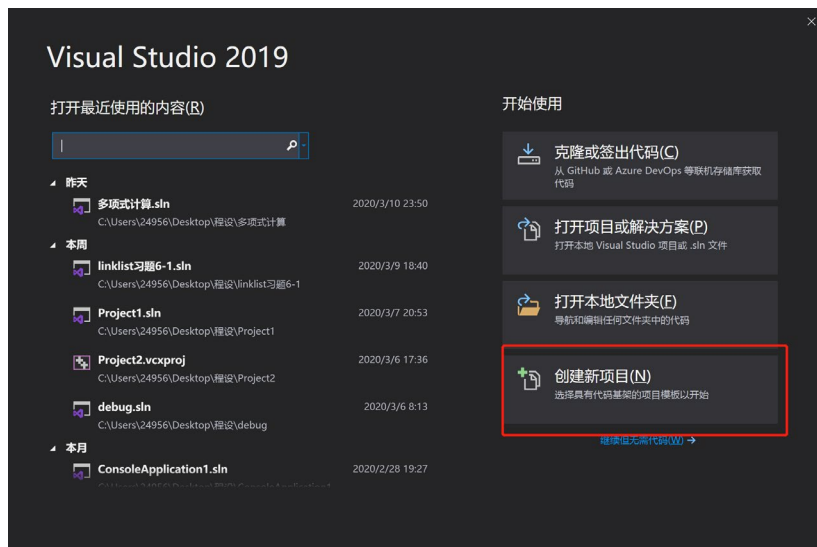


VS2019 使用基础小教程

安装的时候选了 c/c++ 那个就好了，打开 vs2019



这个随便了，我也不知道不勾会咋样

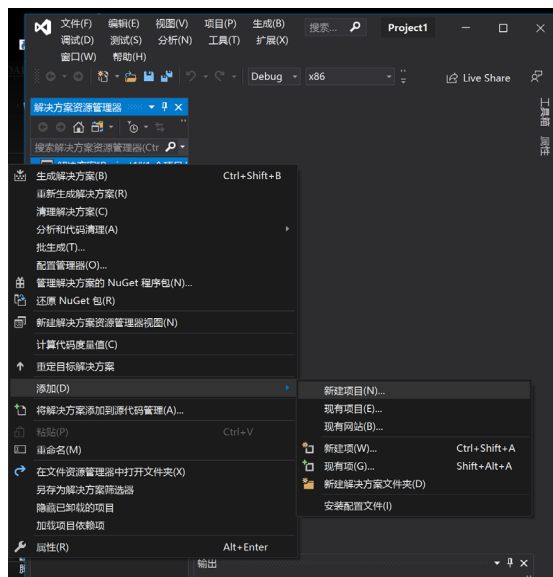
创建之后就在 project1 这个解决方案 solution 的 project1 这个项目里

一个解决方案里面可以有好多个项目

所以要多项目的话可以直接建在这个 solution 里面

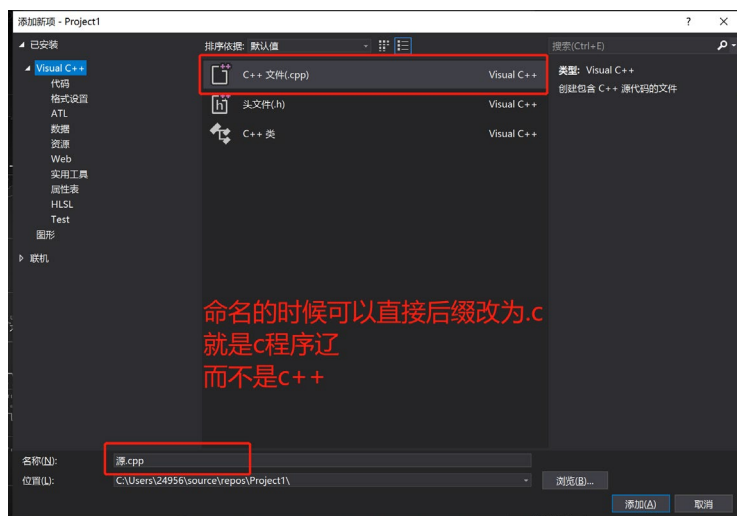
右键解决方案->添加->新建项目就好

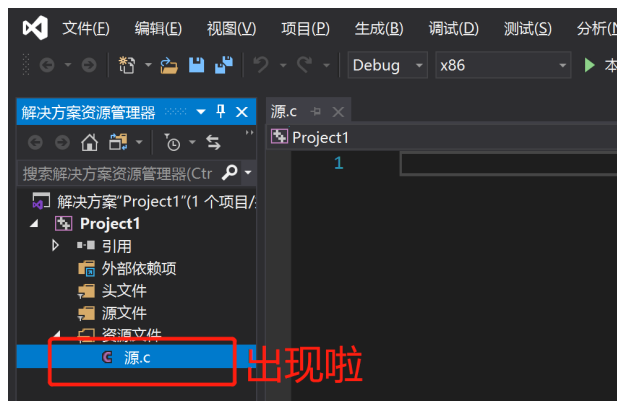
就不用再从头建解决方案辽，新项目还是可以选择空项目或者有 helloworld



如果选了建立空项目，那要开始写代码，代码放在哪里呢

右键“资源文件”->添加->新建项





我写了个简单的 scanf 和 printf

然后是怎么编译运行

生成->生成解决方案/生成 Project1 这一步大概相当于编译?

两个生成的区别:

生成解决方案: 同一解决方案下面的 所有项目 都操作

生成某个 project: 操作当前项目

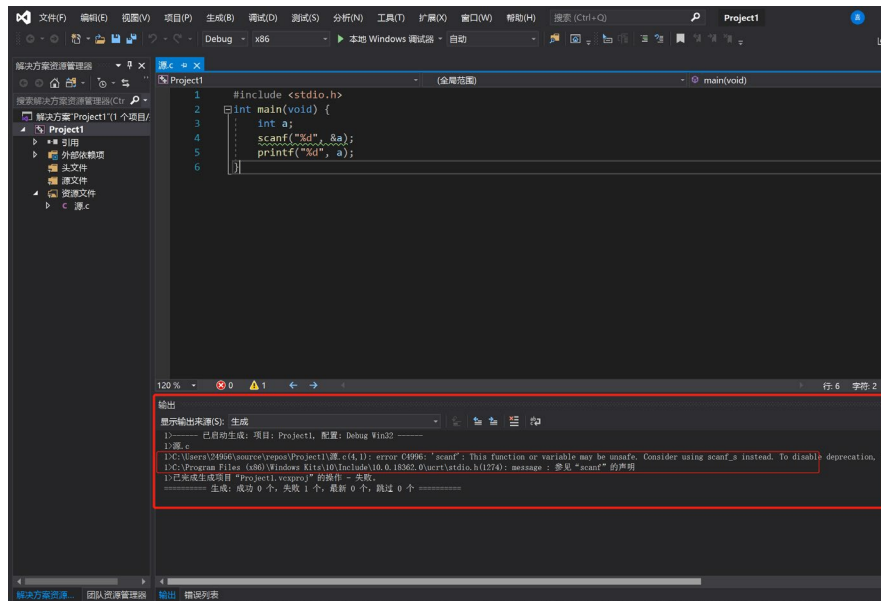
我也不知道重新生成有啥区别作用



这时候会编译失败, 因为 vs 默认用的是 scanf 好像要用什么 wscanf

如果要兼容是残废, 需要进行设置

看界面下方的输出



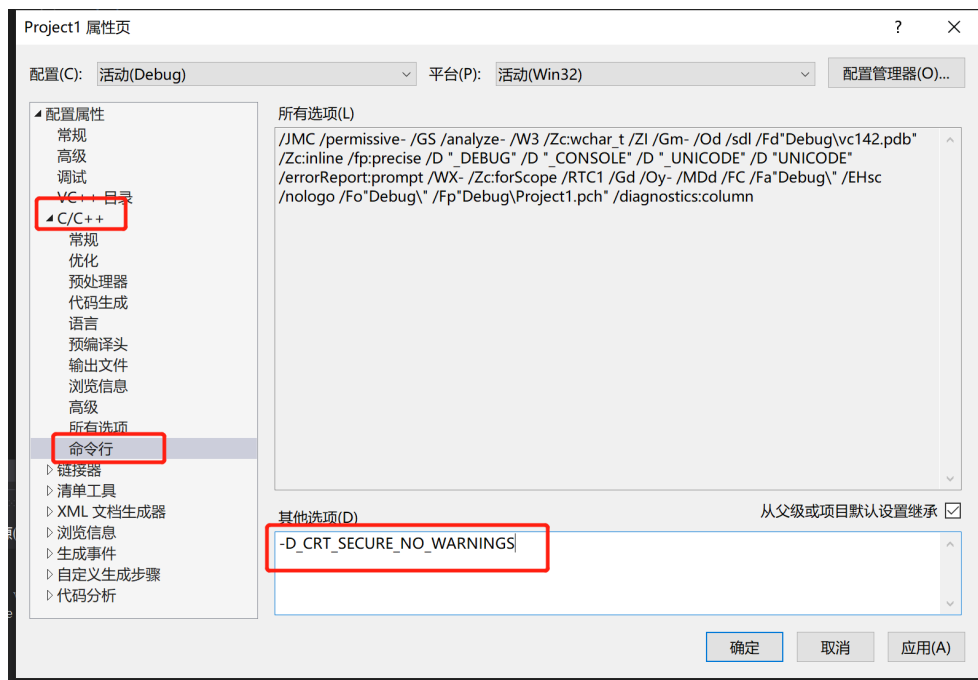
往右边拉进度条，看到有一串东西

To disable deprecation, use `_CRT_SECURE_NO_WARNINGS`.

复制这个一堆 `_CRT_` 的东西

右键项目（也就是 project1）

找到最底下的“属性”



-D 之后把刚刚复制的东西贴进去->确定

回来再生成解决方案/project1, 发现 scanf 不再报错了, 编译成功了

然后运行, 操作:

调试->开始调试



一切照常

可以充当 dev 写写题目了

开头不是说可以一个解决方案下面好几个项目吗

现在看看多项目

右键解决方案->添加->新项目

在新项目里的资源文件添加新项，开始写代码，然后生成解决方案

(这时候就可以感受到两种生成解决方案操作的不一样了)

Scanf 仍然报错，与 project1 一样操作一遍，这个可以同时改一个项目里的设置，但是好

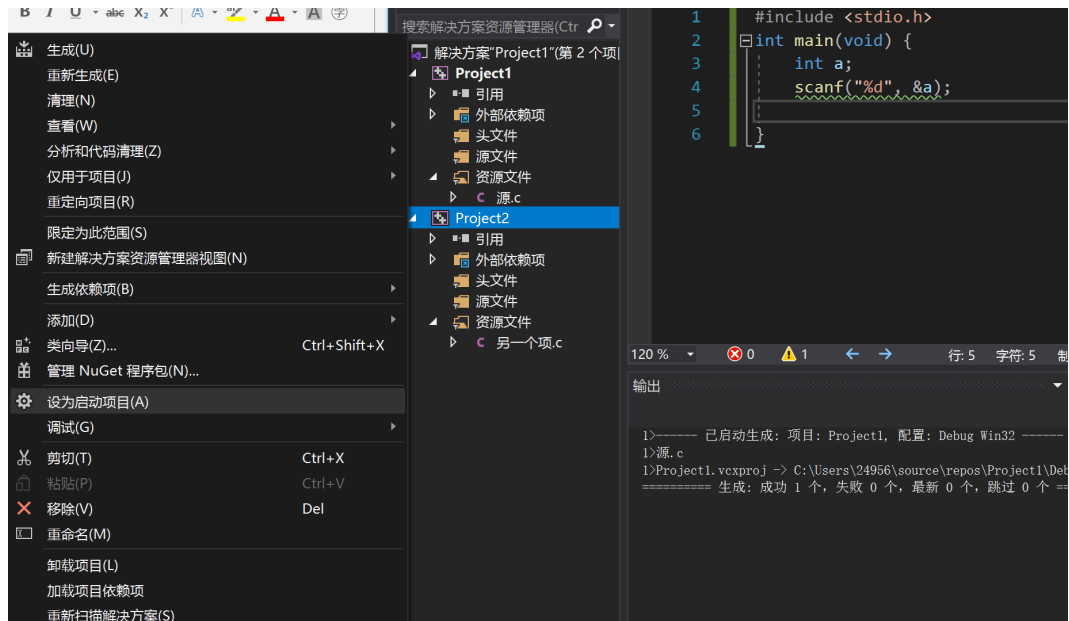
像不能对解决方案统一操作

这时候调试，会发现它仍然运行的是 project1，而没有自动切换到 project2

要调试 project2 的操作：

右键 project2->设为启动项目

然后再调试就 ok 了



然后，断点就和 vscode 一样？点一下那个行数右边就会跳出小红○了

调试应该也差不多？dev 里我没用调试功能

逐语句调试真好啊 还可以看变量变化（由衷感叹