

在Python的交互式命令行写程序，好处是一下就能得到结果，坏处是没法保存，下次还想运行的时候，还得再敲一遍。

使用文本编辑器

所以，实际开发的时候，我们总是使用一个文本编辑器来写代码，写完了，保存为一个文件，这样，程序就可以反复运行了。

现在，我们就把上次的 `hello_world` 程序用文本编辑器写出来，保存下来。

那么问题来了：文本编辑器到底哪家强？

推荐两款文本编辑器：

一个是 [Sublime Text](#)，免费使用，但是不付费会弹出提示框：



一个是 [Notepad++](#)，免费使用，有中文界面：



请注意，用哪个都行，但是**绝对不能用Word和Windows自带的记事本**，Word保存的不是纯文本文件，而记事本会自作聪明地在文件开始的地方加上几个特殊字符（UTF-8 BOM），结果会导致程序运行出现莫名其妙的错误。

安装好文本编辑器后，输入以下代码：

```
print('hello, world')
```

注意**保存**前面不要有任何空格，然后，选择一个目录，例如 `c:\work`，把文件保存为 `hello.py`，就可以打开命令行窗口，把当前目录切换到 `hello.py` 所在目录，就可以运行这个程序了：

```
C:\work>python hello.py
```

```
hello, world
```

也可以保存为别的名字，比如 `first.py`，但是必须要以 `.py` 结尾，其他的都不行，此外，文件名只能是英文字母、数字和下划线的组合。

如果当前目录下没有 `hello.py` 这个文件，运行 `python hello.py` 就会报错：

```
C:\Users\ljj>python hello.py
python: can't open file 'hello.py': [Errno 2] No such file or directory
```

报错的意思就是，无法打开 `hello.py` 这个文件，因为文件不存在。这个时候，就要检查一下当前目录下是否有这个文件了，如果 `hello.py` 存放在另外一个目录下，要首先用 `cd` 命令切换当前目录。

视频演示：

直接运行py文件

有同学问，能不能像 `exe` 文件那样直接运行 `.py` 文件呢？在Windows上是不行的，但是，在Mac和Linux上是可以的，方法是在 `.py` 文件的第一行加上一个特殊的注释：

```
#!/usr/bin/env python
```

```
print('hello, world')
```

然后，通过命令给 `hello.py` 以执行权限：

```
$ chmod +x hello.py
```

就可以直接运行 `hello.py` 了，比如在Mac下运行：



小结

用文本编辑器写Python程序，然后保存为后缀为 `.py` 的文件，就可以用Python直接运行这个程序了。

Python的交互模式和直接运行 `.py` 文件有什么区别呢？

直接输入 `python` 进入交互模式，相当于启动了Python解释器，但是等待你一行一行地输入源代码，每输入一行就执行一行。

直接运行 `.py` 文件相当于启动了Python解释器，然后一次性把 `.py` 文件的源代码给执行了，你是没有机会以交互的方式输入源代码的。

用Python开发程序，完全可以一边在文本编辑器里写代码，一边开一个交互式命令行窗口，在写代码的过程中，把部分代码粘贴到命令行去验证，事半功倍！前提是得有个27寸的超大显示器！

参考源码

[hello.py](#)