

# 安装说明

安装和设置说明都包含在书中，但有时使用带链接的在线说明会更方便。这里的资料还比书中包含更多屏幕截图。

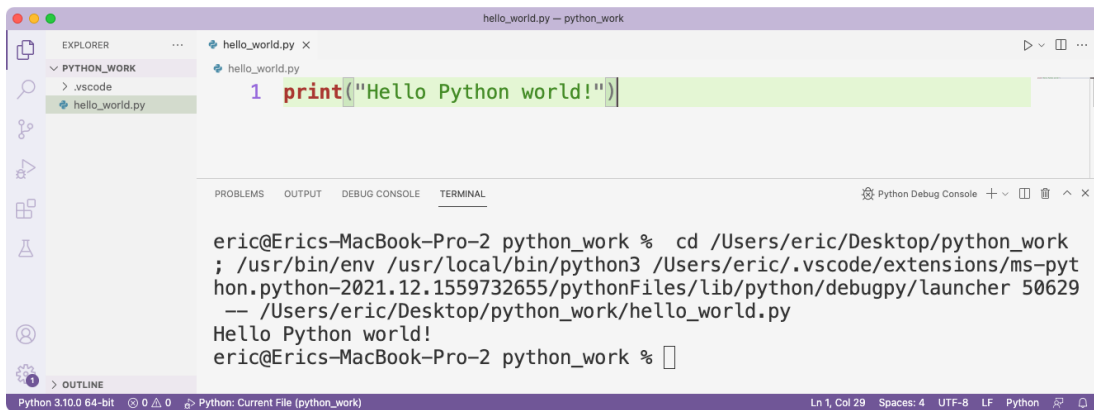
## 配置 VS Code

VS Code 是一款出色的文本编辑器，提供了一些通常仅在 IDE 中才有的特性。附录 B 提供了一些很有用的建议，可帮助你高效地配置和使用 VS Code。这里重复附录 B 介绍的一些配置步骤，而且提供了更多的屏幕截图。

### 简化输出

默认情况下，VS Code 在集成的终端窗口中显示输出。这个终端窗口内嵌在应用程序 VS Code 中，让你无须打开独立的终端窗口就能看到输出。然而，这个窗口显示了很多你在刚学习运行程序时不想看到的内容，因此可能显得有点乱。

例如，当你运行 `hello_world.py` 时，VS Code 界面类似于下面这样：



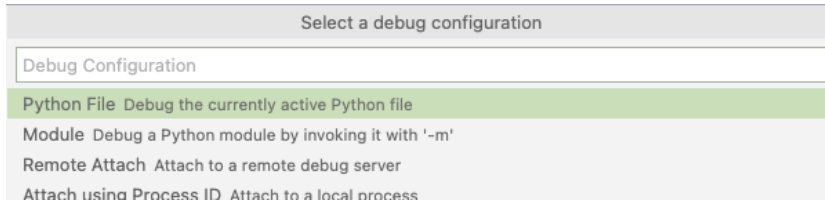
你真正关心的输出只有一行，那就是 `Hello Python world!`。其他的输出指出了文件 `hello_world.py` 的存储位置，以及用来运行程序的 Python 解释器所在的位置。当前，你可能只想看到程序的输出。

要只显示程序的输出，可采取如下措施。

- 关闭所有已打开的选项卡，再退出 VS Code。
- 重新启动 VS Code，并打开你用来存储 Python 文件的文件夹。
- 单击 Run/ Debug（运行和调试）图标：



- 单击 **Create a launch.json File** (创建 launch.json 文件), 再选择 **Python** 并单击包含 **Python File** 字样的选项。

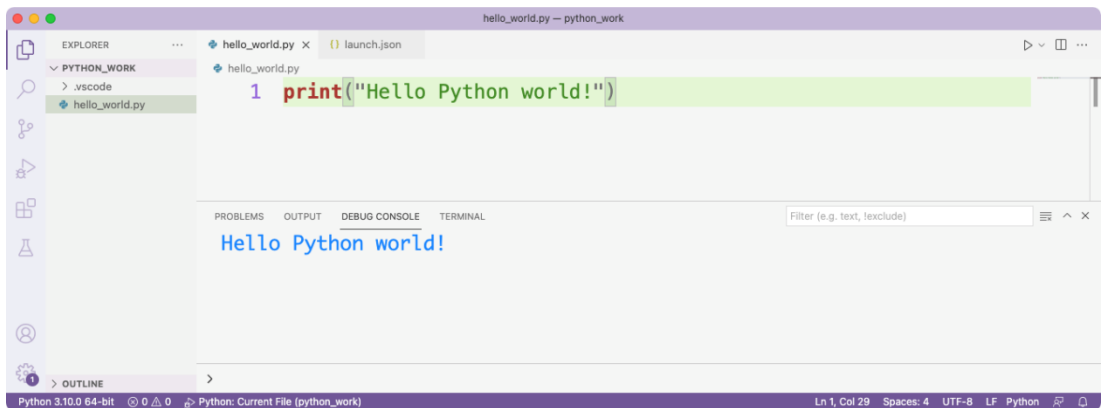


在 `configurations` 部分, 将设置 `console` 的值 `integratedTerminal` 改为 `internalConsole`:

### launch.json

```
{
  ...
  "configurations": [
    {
      "name": "Python: Current File",
      "type": "python",
      "request": "launch",
      "program": "${file}",
      "console": "internalConsole"
    }
  ]
}
```

现在, 再次运行你的 `.py` 文件, 并在底部窗格中单击标签 **Debug Console** (调试控制台), 将发现这个选项卡中只显示了程序的输出。



## 运行使用了 `input()` 的程序

完成上述修改，让 VS Code 在调试控制台（而不是集成的终端窗口）中显示输出后，将无法运行使用了 `input()` 函数的程序。调试控制台是只读的，这意味着它不能接受输入。如果程序使用了第 7 章介绍的 `input()` 函数，需要让 VS Code 在集成的终端窗口中显示输出，以便能够运行这种程序。

为此，可打开前面创建的文件 `launch.json`，并将设置 `console` 的值从 `internalConsole` 改回 `integratedTerminal`：

### `launch.json`

```
{
  ...
  "configurations": [
    {
      "name": "Python: Current File",
      "type": "python",
      "request": "launch",
      "program": "${file}",
      "console": "integratedTerminal"
    }
  ]
}
```

## 在编辑器和终端窗口之间切换

在运行使用了 `input()` 函数的程序时，需要在用于输入代码的编辑器和用于提供输入的终端窗口之间切换。尽管可以使用鼠标或触控板，但使用键盘快捷键更有帮助。

在编辑器中输入代码后，可按 `Ctrl + F5` 来运行程序。在终端窗口中，将出现一个提示符，让你能够给程序提供输入。要从编辑器切换到终端窗口，可按 `Ctrl + ``；提供程序输入后，可再次按 `Ctrl + `` 从终端窗口切换到编辑器窗口。